AFGL-TR-80-0048

LEVEL II

STUDY OF BACKGROUND RADIANCE IN UPPER ATMOSPHERE

Hajime Sakai

Astronomy Research Facility
University of Massachusetts
Amherst MA 01003

DTIC
ELECTE
DEC 1 2 1980
S   D
E

January 1980

Final Report
  1 July 1975 - 30 September 1979

Approved for public release; distribution unlimited.

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

80 12 11 007

Qualified requestors may obtain additional copies from the Defense
Documentation Center. All others should apply to the National
Technical Information Service.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFGL-TR-80-0048	2. GOVT ACCESSION NO. AD-A092 937	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) STUDY OF BACKGROUND RADIANCE IN UPPER ATMOSPHERE.		5. TYPE OF REPORT & PERIOD COVERED Final Rept. 1 Jul 1975 to 30 Sep 79
		6. PERFORMING ORG. REPORT NUMBER UMASS-ARF-80-308
7. AUTHOR(s) Hajime Sakai		8. CONTRACT OR GRANT NUMBER(s) F19628-75-C-0187
9. PERFORMING ORGANIZATION NAME AND ADDRESS Astronomy Research Facility University of Massachusetts Amherst MA 01003		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE61102F 2310G404 & 2310G408
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Geophysics Laboratory Hanscom AFB, Massachusetts 01731 Monitor/Richard Nadile/OPR-1		12. REPORT DATE January 1980
		13. NUMBER OF PAGES 85
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was sponsored in part by the Defense Nuclear Agency
Subtask I25AAXHX632, Work Unit 42, entitled "SPIRE Instrumentation."

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Atmospheric emission
Radiative transfer
Scattering
Infrared

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The final report on the analysis of the SPIRE data. The major CO_2 IR bands are analyzed by using a simplified model.

DD FORM 1473 EDITION OF 1 NOV 68 IS OBSOLETE
1 JAN 73
S/N 0102-014-6601 |

Introduction

The work of this contract was intended to develop the atmospheric model which integrates the emission and the scattering term in a combined form and to provide analysis of the earth atmospheric emission data gathered by the SPIRE payload in these two terms.

Our work performed throughout the entire period can be divided into five areas:

(1) incorporation of various computer programs designed to calculate the atmospheric infrared emission level under a specific condition;

(2) improvement of the algorithm employed in these programs;

(3) development of an atmospheric infrared radiance model which unifies the emission and scattering term;

(4) application and evaluation of the developed model to analysis of the SPIRE data; and

(5) evaluation of some preliminary test data for the coming AFGL balloon-borne atmospheric emission measurement experiment.

The SPIRE experiment[1] was carried out during 1978, and the collected data were made available to us in late 1978. Our effort during the early phase of the contract period, i.e., prior to the SPIRE launching, was focused in the area of (1) and (2). Our Scientific Report No. 1 summarizes our progress made during the early phase.[2] The present report will describe our work covered in the entire period, with a special emphasis on the progress made since the last report.

Upon inspection of the SPIRE data, it became obvious to us that the existing computation programs are much more elaborate than required in the

1

analysis. A quick check on the observed scattering radiance level revealed
that the single-scattering theory with the Rayleigh particles[3,4] produced a
result adequate for the analysis. The Monte Carlo code developed by
Blattner and his co-workers[5,6,7] is found unnecessary for the analysis, because
the computation based on the single scattering process produced a result
which agrees, reasonably well, with the SPIRE observed data. The existing
computation programs treat the scattering process entirely separately from
the absorption-emission process. We find them rather inadequate for
explaining why some molecular bands change their absorptive behavior
against the scattering background to become emissive as the altitude
increases. A theory which unifies both processes in the radiative
formulation is definitely needed for the analysis. A limit placed on time
restricted us to develop a computation algorithm which generates the
spectral radiance level as a function of frequency and an observation
condition in an automatic fashion. We feel that we are on the right track
for seeking a solution to the problem. Nonetheless, we came to understand
the basic radiative mechanism of the upper atmosphere with an extent that
we can place a proper perspective to the analysis of the SPIRE data.

Radiative Process

   The radiative transfer in the atmosphere is traditionally analyzed in
treating the infrared radiation field as a continuous flow of radiative
energy. The interactions between the radiation field and the atmospheric
gas is formulated in an integral equation. Several computer programs have
been written for analysis of the radiation transfer problem. Our approach
taken in the present study is slightly different from the traditional
analysis. We tried to analyze the radiative transfer in the upper atmosphere
in following the interactive process between the photons and molecules.

2

The photons incident into a layer of atmosphere would go through one of three processes: (1) they may be absorbed by the atmospheric molecules or atoms; (2) scattered by the Rayleigh particles or by the aerosols; and (3) left unperturbed. We can formulate these interactions between the photon field and the molecular system in terms of the density of these two interacting partners and a quantity which indicates a strength of the interaction.

The interaction between the photons and the molecules (or the atoms) in the absorption process is specified by Einstein's formulation.[8] The number of molecular transitions N_t in unit time per unit volume, when interacting with the photons of a frequency ν, is given by

$$N_t = (N_\ell B_{\ell u} - N_u B_{u\ell})\rho(\nu). \qquad (1)$$

In this expression, the following notations are used:

N_ℓ: number of molecules per unit volume in lower state ℓ

N_u: number of molecules per unit volume in upper state u

$B_{\ell u}$:   Einstein coefficient for induced absorption

$B_{u\ell}$:   Einstein coefficient for induced emission

$\rho(\nu)$:   spectral density of the radiation field in unit volume at frequency ν

The density $\rho(\nu)$ is related to the photon density N_p given in unit volume per second by

$$\frac{c}{4}\rho(\nu) = N_p h\nu, \qquad (2)$$

where h is the Planck constant and c is the speed of light. Combining Eqs. (1) and (2), the molecular transitions N_t is given by

$$N_t = (N_\ell B_{\ell u} - N_u B_{u\ell})\frac{4 N_p h\nu}{c}. \qquad (3)$$

This expression can be rewritten by using the molecular absorption

3

strength $S_{\ell u}$ (cross-section)[9]

$$N_t = 4N_p N_o S_{\ell u},\tag{4}$$

where

$$S_{\ell u} = \frac{h\nu(N_\ell B_{\ell u} - N_u B_{u\ell})}{N_o c} \simeq \frac{h\nu}{c} B_{\ell u}\left(1 - \frac{N_u}{N_\ell}\right)\tag{5}$$

with N_o the number of molecules in unit volume. In deriving Eq. (5), it is assumed that

$$N_o \simeq N_\ell.\tag{6}$$

The Eq. (4) derived above specifies the number of molecular transitions N_t in the photon density N_p, the molecular density N_o, and the molecular absorption strength $S_{\ell u}$ which is experimentally determined in the laboratory.

The scattering process can be formulated in the same way.[4] A traditional approach is somewhat different in that the incident radiation field is given in the irradiance; the power scattered toward the direction ϕ by a molecule is given by

$$\sigma_m(\phi)E_o = I(\phi),\tag{7}$$

where E_o is the irradiance of the incident plane wave. We can formulate the incident irradiance by

$$E_o = N_p h\nu,\tag{8}$$

again using N_p, the photon density in unit volume per unit second. Now we find that the power scattered by a molecule is given by

$$I(\phi) = N_p h\nu\, \sigma_m(\phi).\tag{9}$$

The radiative power scattered toward the direction ϕ per unit solid angle by an ensemble of molecules N_o in unit volume is given by

$$I'(\phi) = N_o h\nu\, N_p \sigma_m(\phi).\tag{10}$$

Thus we get the desired form for the number of photons scattered toward the

4

direction ϕ;

$$N_\rho(\phi) = N_o N_p \sigma_m(\phi). \tag{11}$$

For the Rayleigh scattering, the scattering cross-section is given by

$$\sigma_m(\phi) = \frac{9\pi^2 \nu^4}{N^2 c^4} \left(\frac{n^2-1}{n^2+2}\right)^2 \sin^2 \phi \tag{12}$$

$$\simeq \frac{4\pi^2 \nu^2}{N^2 c^2} (n-1)^2 \sin^2 \phi$$

where N is the number of molecules in a unit volume at S.T.P.[4]

The energy absorbed by a molecule from the radiative field is sooner or later released from it in a de-excitation process. In the upper atmosphere, the de-excitation is done by the radiative process. The photon absorbed by a molecule is re-emitted to the radiation field isotropically. If the original absorption occurs between the ground state and the first excited vibrational state, there is no significant change in the photon energy of the absorbed and of the re-emitted. In the lower atmosphere, a local thermal equilibrium is established; the energy absorbed by a molecule is dissipated into the kinetics energy shared in the molecular ensemble. The interaction between the molecular ensemble and the radiation field under the local thermal equilibrium condition is treated below in calculation of the photon density. The de-excitation under this condition is done predominantly by the collisional process. In a transient region from the lower to the upper atmosphere, the molecular system de-excites by both processes.

Photon Density in Atmosphere

There are two radiation sources which must be considered in the infrared

radiative process of the atmosphere: the solar radiation penetrating the
atmosphere from the top, and the radiation going upward to escape to the
outerspace from the earth's surface. The energy density $\rho(\nu)d\nu$ of the
black-body at temperature T is given by

$$\rho(\nu)d\nu = \frac{8\pi h\nu^3}{c^3} \frac{d\nu}{e^{\frac{h\nu}{kT}}-1} . \tag{13}$$

The photon density N_p in unit volume per unit second per wavenumber interval
$d\sigma$ emitted by the same blackbody radiator is

$$N_p \, d\sigma = \frac{c^2}{4h\nu} \rho(\nu)d\nu$$

$$= \frac{2\pi\nu^2}{c} \frac{d\sigma}{e^{\frac{h\sigma c}{kT}}-1} = \frac{2\pi\sigma^2 c \, d\sigma}{e^{\frac{hc\sigma}{kT}}-1} . \tag{14}$$

For the telluric radiation, the photon density is calculated by taking T at
a value between 225°K and 300°K. For the solar radiation, it is calculated
by the same expression with an attenuation factor;

$$N_{p_\odot} = \left(\frac{r_s}{R_s}\right)^2 \frac{2\pi\sigma^2 c}{e^{\frac{hc\sigma}{kT}}-1} , \tag{15}$$

where r_s is the solar radius and R_s is the solar distance, T is about 6000°K.
The front factor $(r_s/R_s)^2$ is approximately 2.2×10^{-5}. In Figure 1, the
photon density for three cases is plotted as a function of the wavenumber σ:
(1) the solar radiation calculated by Eq. (15), (2) a blackbody at T = 300°K,
and (3) a blackbody at T = 225°K. The telluric photon density in the upper
atmosphere is affected by the optical thickness of the lower atmosphere. In
some spectral ranges, there is no atmospheric molecular absorption, and the
photon density for the 300°K blackbody remains unaffected from the earth's
surface to the top of atmosphere. In a region where the atmospheric

absorption is extreme, the photon density in the upper atmosphere is controlled by the energy density of the blackbody radiation at the tropopause temperature T = 225°K. We can observe in Figure 1 that the molecules and the atoms in the upper atmosphere are excited by two types of radiations which have a vastly different spectral characteristic. The telluric photon density remains unaffected by the solar daily motion. The photons in a region below 2000 cm^{-1} are predominantly telluric, while those in a region above 2000 cm^{-1} in the daytime are solar.

The emission due to the CO_2 transitions, one for the $(01101-00001)$[10] band at 670 cm^{-1} (the ν_2 band) and another for the $(00011-00001)$ band at 2300 cm^{-1} (the ν_3 band), is ready for a qualitative analysis using the photon density calculated for the curves shown in Figure 1. (A more detailed quantitative analysis will follow later in the report.) Once accepting the photon spectrum shown in Figure 1, we find that the photons available for exciting the ν_2 band are telluric. The temporal characteristics of the telluric photons remain independent of the daily cycle. Thus the ν_2 $(01101-00001)$ band emission at 670 cm^{-1} must remain unchanged between the day and the night time. The SPIRE data shown in Figure 2 for this CO_2 band confirms a stationary nature of the emission. The CO_2 ν_3 $(00011-00001)$ band at day side is excited by the solar photons, and by the telluric photons at night side. Since the ν_3 transition is very strong, the telluric photons available in the upper atmosphere are those of a 225°K blackbody. The difference in these two densities between the solar and the telluric is two orders of magnitude. Thus the radiance level of the transition should undergo a change of about 2 orders of magnitude between the day and the night. The SPIRE data in Figure 3 indicates that the ν_3 band emission primarily undergoes the diurnal change described above. Upon a careful inspection of the data, the radiance level at nighttime is brighter than what can be predicted by the telluric photon level.[17] We will make an analysis on this anomaly later in the report.

7

Figure 1

8

Figure 2

9

Figure 3

10
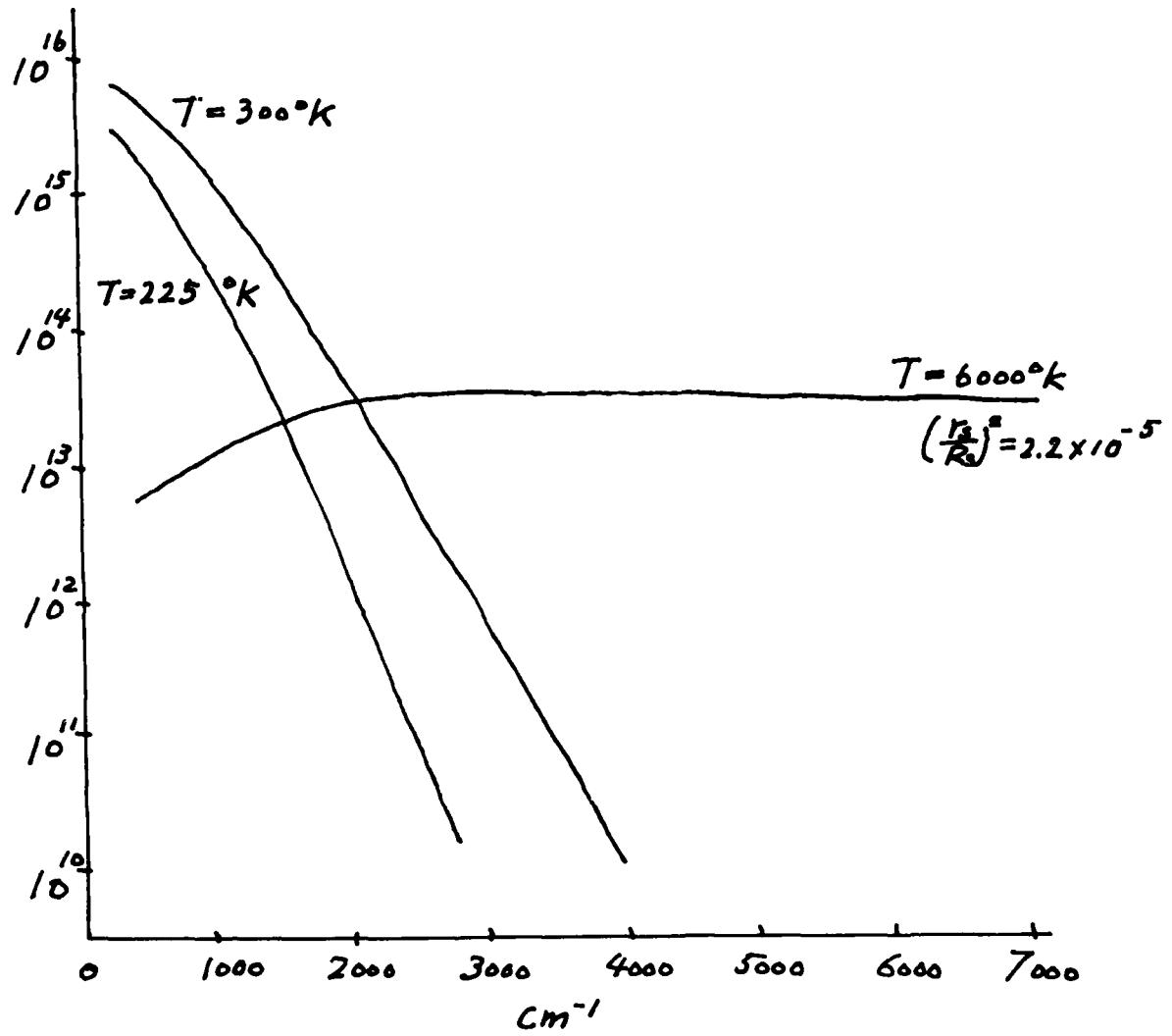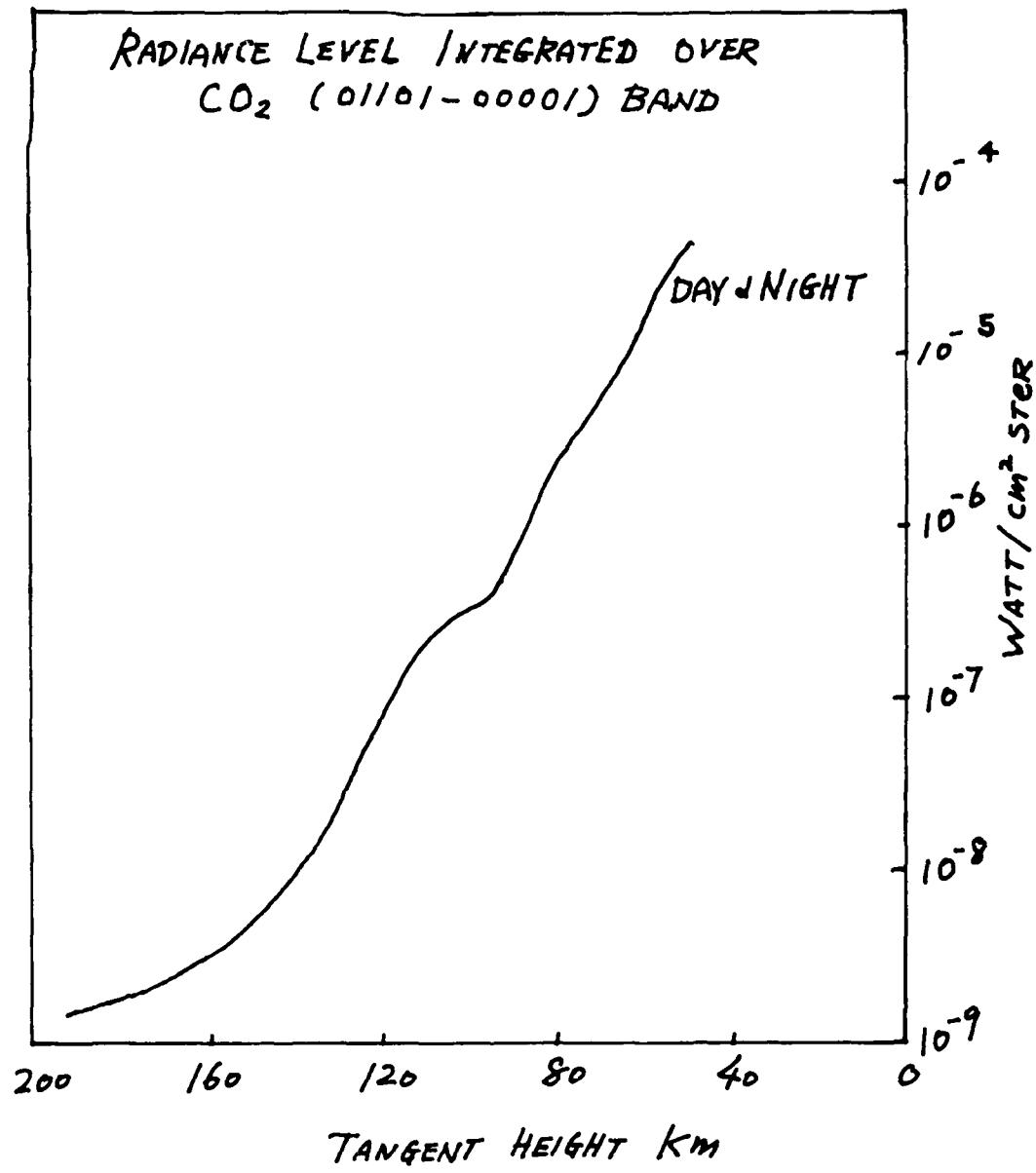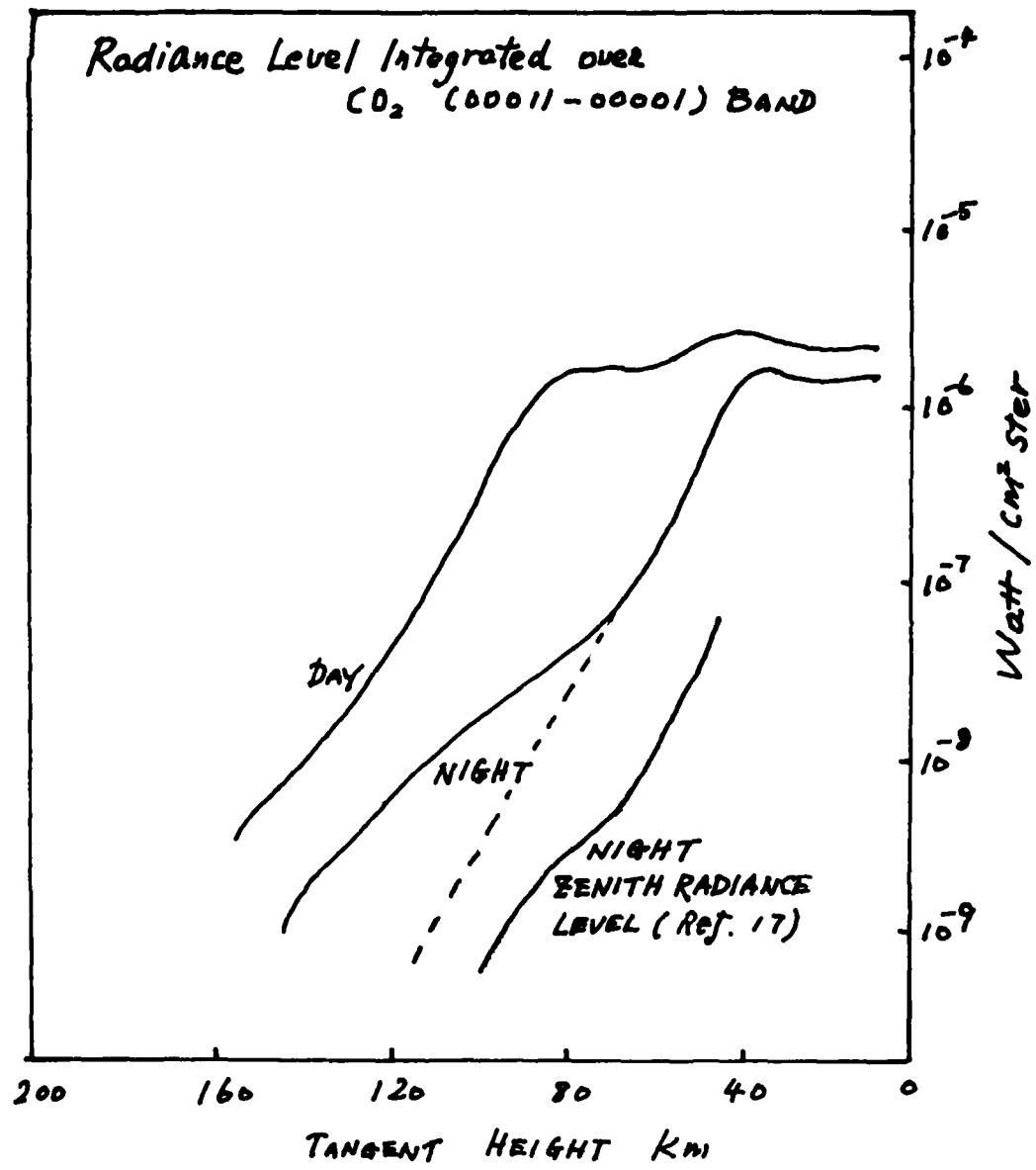
Model for Calculating Atmospheric Radiance Level

The expressions obtained above in Eqs. (4) and (11) are concerned with an individual interacting process in which a single photon is either absorbed or scattered by a molecule. In this section we will develop a model which relates the atmospheric radiance level to the basic interaction process given by these expressions.

The energy absorbed by a molecule in interaction with the photon field must be released in a de-excitation process. In the upper atmosphere, the only process available for de-excitation is a radiative process. A question which must be raised and answered for such a radiative excitation-de-excitation process concerns an establishment of equilibrium population density among various molecular states. Degges and other workers developed the computation program capable of giving an estimate of the radiative level in the normal atmospheric environment at high altitude.[11,12] We are truly thankful for the existence of these programs which provided a valuable help to develop our insight into the problem.

We modified the Degges program to suit our needs, and to improve the computation efficiency in our CYBER system. A brief description of our effort along this direction will be given later in this report, together with the listing of our modifications. The atmospheric model is judged on its ability for producing a reasonable estimate of the infrared radiance level; in particular, it is questioned on precision and practicality.

The computation scheme to obtain the radiance level must find the equilibrium population distribution which is established among the vibrational states when the molecular system is subjected to balance with the photon field. The precision needed in the computation is somewhat controlled by a spectral resolution of the result. The emission involving

11

the hot-band transition can be distinguished only with an improved spectral resolution, because it is usually buried under the fundamental transition. We decided to ignore these hot-band transitions from our model in a belief their contribution is not significant to the computation. The assumptions made in our model* are certainly open for debate and all criticisms to our viewpoint are welcome. Our view that all transitions observed in the data occurred from the ground state, in turn, simplified the calculation to a great extent. A direct correlation was resulted in our formulation between the radiance level for a particular tangent height and the molecular density of the concerned molecular system at this altitude.

The total energy emitted by the molecules in a unit volume through the radiative de-excitation is equal to the energy absorbed by the same molecules if that is the only process feasible. The absorbed energy is given by the number of transition times the photon energy: $b' = h\nu N_t$. The energy released in the radiative de-excitation is emitted isotropically. Thus the energy re-emitted per a unit area over a unit solid angle is given by: $b = b'/4\pi = h\nu N_t/4\pi$. The photons released from a molecule are very unlikely absorbed again, since the atmosphere is optically thin.[†] Even if they are absorbed, they are re-emitted in the photon field. The photons which reach a sensor of a unit area with a field view of a unit solid angle are computed simply by accumulating the contribution from each unit volume which falls into a line of sight. Since the optical étendu preserves, the summation which we must take in the calculation is equal to that of each contribution b taken over a column of a unit cross-section extending from the sensor to the space all the way along the line of sight. The radiance level per unit solid angle integrated over the band is then given by

---

*See the Appendix D.
†The absorption along the limb-view path of 100 km tangent height is less than 5% for the (00011-00001) band of CO_2.

12

$$B = \frac{h\nu N_t}{4\pi} = \frac{h\nu S_{u\ell} \, N \, N_p}{\pi} \qquad (16)$$

where N is the total number of molecules counted along the line-of-sight per unit area;

$$N = \int N_o \, d\ell \; . \qquad (17)$$

The density of atmospheric molecules falls off approximately exponentially with the altitude. If we can assume the density $N(h)$ as a function of altitude h given by

$$N(h) = N(h_o) \, e^{-\frac{h-h_o}{H}} \qquad (18)$$

with a characteristic constant H, the total number of molecules between two altitudes h and h_o is given by

$$N(h,h_o) = \int_{h_o}^{h} N(h) \, dh = N(h_o) \int_{h_o}^{h} e^{-\frac{h-h_o}{H}} \, dh \qquad (19)$$

$$\simeq N(h_o) \, H$$

For the limb view calculation, we need to calculate a distance ℓ between two altitudes, the tangent height x and a height above by H. The distance ℓ is calculated by

$$\ell = (x+H+\rho) \sqrt{1 - (\frac{x+H}{x+H+\rho})^2} \; \simeq \; \sqrt{(2H)\,(x+\rho)} \qquad (20)$$

where ρ is the earth radius ($\simeq$ 6360 km). To calculate the total density N along a line of sight to the tangent height x, we use the approximation Eqs. (19) and (20):

$$N = N_o(x) \, 2 \sqrt{2(H)\,(x+\rho)} \qquad (21)$$

where $N_o(x)$ is the molecular density at altitude x. With the approximate total density N derived in Eq. (21) the limb-view radiance at the tangent

height x, is given by an expression

$$B = \frac{h\nu}{\pi} S_{u\ell} \, N_o(x) \; 2 \; \sqrt{2H(x+\rho)} \; N_p. \tag{22}$$

The expression given by the equation above requires an estimate of the photon density N_p. The value shown in Figure 1 is given in a unit wavenumber interval. The photons which contribute to the transition are those within the width of the vibrational-rotational lines. If we take the photon density to be the value over the band, say 100 cm^{-1} for the CO_2 (01101-00001) transition, we obviously get an erroneous result. The effective photon density must be adjusted by multiplying an effective number of the rotational lines within the band times the line width, which we can safely assume the Doppler width for those in the upper atmosphere.

CO_2 Bands, (01101-00001) and (00011-00001)

The CO_2 molecule is relatively stable in the atmosphere, because neither electronic transitions nor photo-dissociation occurs in the entire spectral range below the Schuman-Runge Continuum of O_2. The mixing ratio remains constant in the atmosphere below 120 km where the O_2 absorption is very effective in blocking those high energy photons in the extreme uv region.

As a result, the equilibrium population distribution is established among various vibrational levels in the electronic ground state with insignificant perturbation from higher excited electronic states. The equilibrium is achieved in balance with the infrared photon field. Under this condition a major concern must be directed to the equilibrium population established for the metastable (10001) and (10002) level, which locates approximately 1300 cm^{-1} above the vibrational ground level. Since there are no optical transitions allowed between those and the ground level, a primary

14

candidate for pumping these levels is the radiationless collisional process which occurs only in the lower atmosphere. The SPIRE data indicate that the emission by the (10001-01101) and (10002-01101) transitions is definitely observable at low tangent height in both the day and night conditions. At high tangent height both of the emissions become less distinctive. It may be concluded that the radiative pumping mechanism for these metastable states is rather minor. The observation coincides with the results obtained by the Degges' radiation balance computation program.

The limb-view radiative levels of the observable CO_2 bands for high-tangent height are calculated with a reasonable accuracy by the model developed above. There are two clearly observable CO_2 emissions in the SPIRE data, one for the (01101-00001) band and another for the (00011-00001). As mentioned above, the emissions by the (10001-01101) and the (10002-01101) transitions are observable in the low-tangent-height data. Both the (10011-00001) and the (10012-00001) band will be discussed together with the scattering continuum observable in the 3000 cm^{-1} ~ 10,000 cm^{-1} region.

The radiance levels shown in Figures 2 and 3 are extrapolated from the SPIRE data by integrating them over the entire band. At a tangent height of 100 km, the (01101-00001) emission is 3×10^{-7} (watt/cm^2 sterad). Using the expression given in Eq. (22):

$$B = \frac{h\nu \ S \ N_o N_p}{\pi}$$

our calculation yields 2.0×10^{-7} watt/cm^2 sterad. In deriving the value, we take

$$h\nu = 1.32 \times 10^{-20} \ \text{Joule}$$
$$S = 8.3 \times 10^{-8} \ \text{mol}^{-1}/cm^2 \ cm^{-1}$$
$$N_o = 5 \times 10^7 \times 10^9 \ \text{for the 100 km tangent height, and}$$

15

$$N_p = 3 \times 10^{15} \times \Delta\sigma \times n = 3 \times 10^{15} \times .001 \times 40,$$

the Doppler width $\Delta\sigma$ calculated by

$$\Delta\sigma = \sigma_o \frac{\sqrt{3kT}}{mc^2} \approx 0.001 \text{ cm}^{-1} \text{ for } \sigma_o = 670 \text{ cm}^{-1},$$

and 40 rotational lines as those effectively emitted.

The SPIRE data indicate that the (00011-00001) emission at 100 km tangent height is 3×10^{-7} (watt/cm^2 sterad) for the daytime level and 3×10^{-8} (watt/cm^2 sterad) for the nighttime level. Our simple-minded calculation shows that $B = 2.1 \times 10^{-7}$ (watt/cm^2 sterad) for the day data and 2.1×10^{-9} (watt/cm^2 sterad) for the night data. In deriving these figures, we take

$$h\nu = 4 \times 10^{-20} \text{ Joule}$$

$$S = 9.6 \times 10^{-17} \text{ mol}^{-1}/cm^2 \text{ cm}^{-1}$$

$$N_o = 5 \times 10^7 \times 10^9$$

$$N_p = 3 \times 10^{13} \times .1 \text{ for the day and } 3 \times 10^{11} \times .1 \text{ for the night.}$$

The observed and the calculated data for the day level agree reasonably well. The observed nighttime radiance level is about one order of magnitude higher than our calculation. In Figure 3, a dotted curve shows the nighttime radiance level calculated by our formulation, Eq. (22). Kumer et al[17] analyzed the nighttime zenith radiance data of the same band (00011-00001), which is shown in the same Figure. Both nighttime calculations are remarkably similar. The SPIRE data differ from the data used in the Kumer's analysis in that they are for the limb view. The total CO_2 molecules involved in both data differ by an order of a magnitude. Thus, the radiance level differs by the same magnitude. The deviation from the exponential decay line in the radiance level occurs approximately at the same altitude for both data. Kumer et al interpreted this anomaly in terms of the extra pumping produced by the vibrationally excited N_2 molecules. The SPIRE data indicates that the

16

difference of the nighttime radiance level from the earthshine figure is more pronounced in the high altitude region where the interaction to the neighbor molecules should decrease. The anomaly in the nighttime radiance level for this band remains mysterious. We studied a possibility of observing the emission by the $W^3\Delta u - B^3\pi g$ transition of N_2 which could be more intense at the altitude above 100 km because of a higher electron density there. A radiance level for the transition estimated on a basis of the known data fails to support the possibility.

Since at a low tangent height the band is completely saturated and the line structure is lost, the photon density available for the absorption is no longer adjusted by the effective number of the rotational lines excited in the band times their width. It is given by the width of the entire band times the photon density: $200 \times 3 \times 10^{11}$ photons/cm^2/sec. Taking the photon energy $h\nu = 4.6 \times 10^{-20}$ Joule, we can estimate that the saturation level is given by

$$\frac{N_p h\nu}{\pi} = 0.9 \times 10^{-6} \ (\text{watt/cm}^2 \ \text{sterad}).$$

Scattering Radiance Observable Between 6000 cm^{-1} and 7500 cm^{-1} and the CO_2-H_2O Band at 3600 cm^{-1}

The radiance level in a spectral region above 4000 cm^{-1} is strongly influenced by the scattering process, in particular the level observable at a low tangent height. For the Rayleigh process the scattering differential cross-section $\sigma(\phi)$ is given by Eq. (12):

$$\sigma_m(\phi) = \frac{9\pi^2\sigma^4}{N^2}\left(\frac{n^2-1}{n^2+1}\right)^2 \sin^2\phi \simeq 4\pi^2\sigma^4\left(\frac{n-1}{N}\right)^2 \sin^2\phi,$$

where σ is the optical frequency in cm^{-1}, both n the refractive index of the atmosphere and N the number density of the air molecule are those values at the S.T.P. condition, and ϕ is the scattering angle. The factor $\left(\frac{n-1}{N}\right)$ is a slow varying function of σ. The scattering cross-section is thus predominantly controlled by σ and ϕ. Figure 4 shows the front factor $4\pi\sigma^2(n-1/N)^2$ as a function of σ. The radiance level obtained by the SPIRE experiment is compared with the computed value for the Rayleigh scattering in Figure 5. The observed values are replotted as a function of σ and of a tangent height in a unit of watt/cm^2 strad cm^{-1}. Irregular structures which can be considered as the molecular absorption or emission are removed by smoothing out. In the computed values, N_p is assumed by the solar value, 3×10^{13} photons/cm^3/S/cm^{-1}. The observed radiative level supports the computed molecular scattering level even though a minor difference exists, dependent on the tangent height and the spectral frequency.

The comparison shown in Figure 5 leads us to conclude two characteristic features of the radiance level in the 3000 ∿ 7500 cm^{-1} range: (1) the scattering radiance is controlled by the single-scattering process; and (2) its main mechanism is the Rayleigh process. The aerosol particles contribute to the over-all scattering process, not significantly even at a low tangent height of 15 km.

18

A problem which we will discuss below plays a central importance in calculation of the infrared radiance level in the upper atmosphere. It is a combined effect of the molecular absorption-emission process and the scattering process. It was treated rather lightly, if not completely side-stepped. The computation modeling of the atmospheric radiance developed to cover a region where only one of these processes is important in the radiative process. There were no models developed to treat the situation where the processes are competitive. The data obtained by the SPIRE experiment is in fact the first to observe the radiance level generated by these processes in competition. The data revealed an extremely remarkable feature for the radiance level in the region of the H_2O-CO_2 2.7μ band (3600 $\sim$ 3800 cm^{-1}). The radiative level observed in the 15 km tangent height data shows an absorptive feature for the band, while that in other data shows an emissive feature. The radiance level remains rather stationary at 10^{-7} (watt/cm^2 strad μ) or 7.5 x 10^{-11} (watt/cm^2 strad cm^{-1}) over a wide range of the tangent height from 15 km to 41 km.

By means of the molecular absorption process the molecule is excited to a higher vibrational state in interacting with the radiation field. The energy absorbed by the molecule in the excitation to a higher vibrational state is equal to that supplied by the photon field. During the de-excitation stage, the same amount of energy is released. Dependent on the environmental surround to the molecule, the energy release takes place to the photon field or to others. If the interaction with the photon field dominates the de-excitation process, there is no net energy loss in the photon ensemble. The absorbed energy from the photon field is released back to the photon field by the de-excitation process. When the collisional process dominates the de-excitation process, the energy absorbed from the photon ensemble is

not released back. The photon density in the ensemble decreases until the photon ensemble reaches a thermal equilibrium with the molecular ensemble. The photon density which is established under a thermal equilibrium is given by the expression which was used to compute the values for Figure 1:

$$N_p = \frac{2\pi\sigma^2 c}{e^{\frac{h\sigma c}{kT}}-1} \quad .$$

The radiative level observable from such an ensemble of the photons and the molecules is given by

$$I = \frac{h\nu N_p}{\pi} \quad .$$

The stationary emission observed for the 2.7μ H_2O-CO_2 band corresponds to the situation described above. What we can observe is the photons generated by the H_2O-CO_2 band which are in a thermal equilibrium at the tropopause temperature. Using the blackbody photon density N_p $.92 \times 10^7$ at 3600 cm^{-1} and $225°K$, and the photon energy $h\nu$ 7.1×10^{-20} Joule at 3600 cm^{-1}, the saturation level I is calculated as 2.1×10^{-11} watt/cm^2 sterad cm^{-1}. With assuming $T = 260°K$, the computation yields 5×10^{-11} watt/cm^2 sterad cm^{-1}. This value is independently determined from the internal excitation-de-excitation mechanism of the molecular system. Whenever the molecular system establishes a thermal equilibrium with the environment under the saturation absorption condition, the photon field must be in an equilibrium with it. The stationary photon flux observed for the 2.7μ H_2O-CO_2 band represents this situation, as was the case for the CO_2 (00011-00001) transition at lower altitude. If the saturation photon flux is larger than the scattering photon flux, the radiance level of the band in question is emissive. If it is smaller, the radiance level is absorptive.

20

Figure 4

21

Figure 5

22
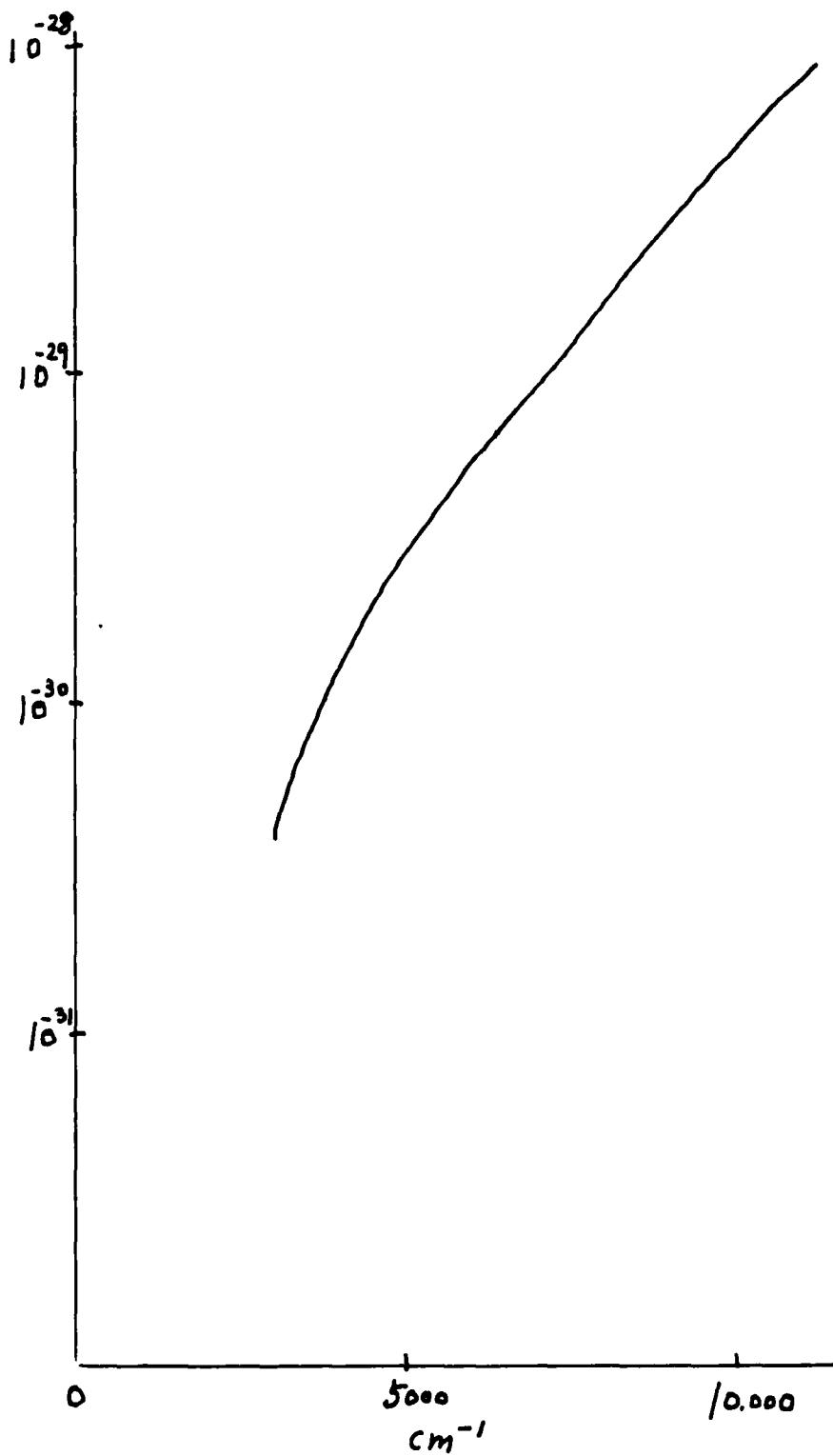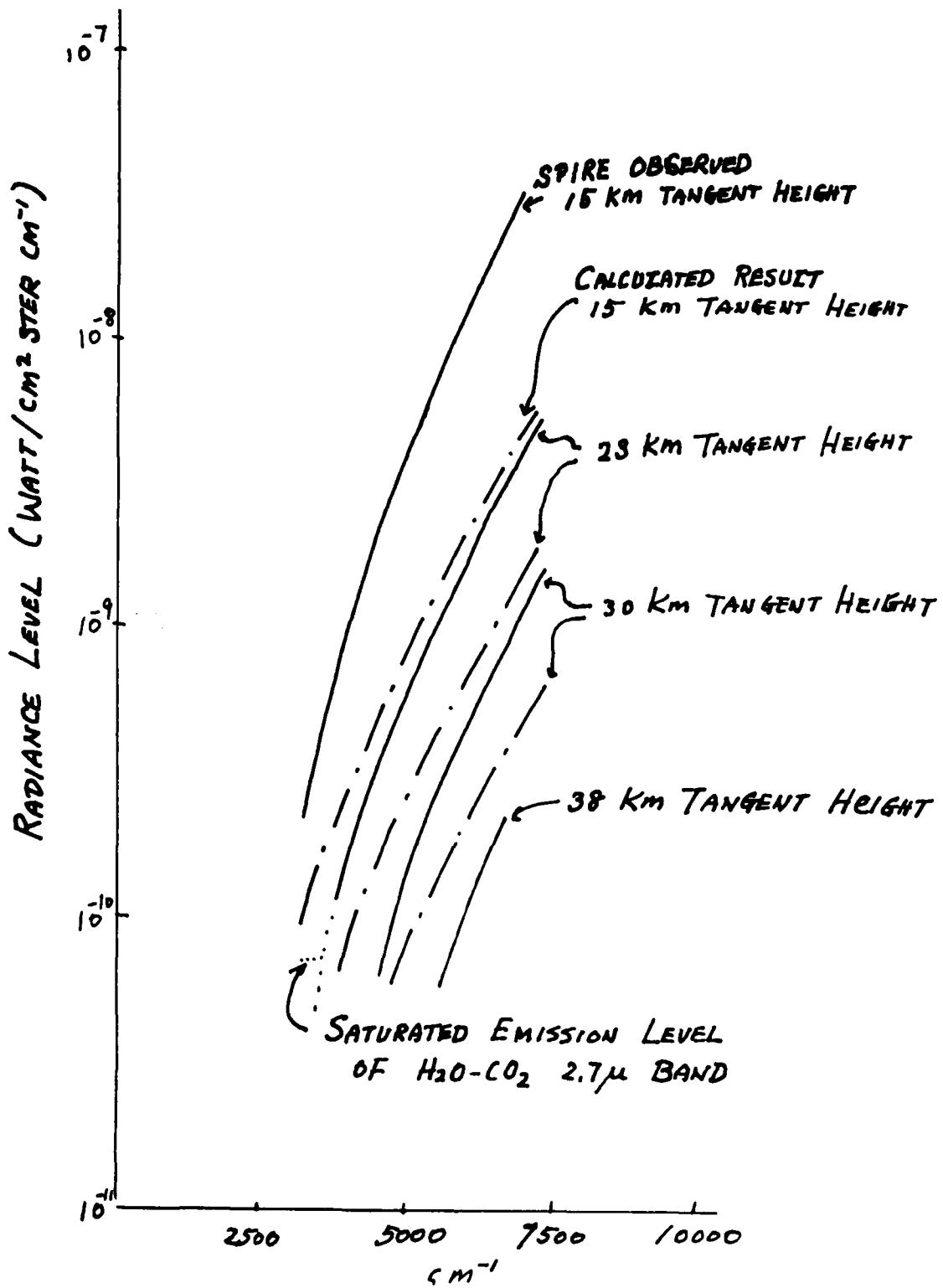
In both cases, the radiation level remains constant.

The Radiance Level of a Molecular Band

In the last two chapters, we discussed the saturated radiance level of the ν_3 band of CO_2 and of the 2.7μ H_2O-CO_2 band. We found that the photon flux under the saturation conditions is given by the blackbody radiation formula; i.e., that the molecular system and the photon ensemble come to establish a thermal equilibrium. For a given molecular band, there exists a critical altitude that the photon density is completely specified by the blackbody radiation formula independent from the vibrational-rotational transition. In the atmosphere above the critical altitude, the photon density is specified by the molecular transition equation, Eq. (4):

$$N_t = 4\ N_o N_s S.$$

A sensor placed very high in the atmosphere collects the photons generated by the molecular transitions which occur along a line-of-sight. With the approximation of a thin optical density air, the total molecular transitions which contribute to the observation are given by integrating the transitions N_t along the path:

$$N = \int N_t \ d\ell \ . \tag{23}$$

Under the approximate expression derived for an observation at a tangent height x, we calculate the total density per a solid angle per a unit cross-section by

$$N(x) = \frac{4N_p S}{\pi} \ N_o(x) \ 2 \ \sqrt{2H(x+\rho)} \ . \tag{24}$$

In deriving the equation above, an assumption is made that the photon density N_p responsible for exciting the molecular transition is constant along the path. In a spectral region below 2000 cm^{-1} where the telluric photons at the tropopause temperature control the excitation, the assumption is obviously

well justified. In a higher cm^{-1} region where the solar photons are dominant, a blackbody temperature corresponding to the solar photon density is much higher than the tropopause temperature as seen in Figure 6. The radiance level of the molecular emission reaches a maximum plateau at the saturation photon density specified by the temperature of tropopause. Thus the molecular transitions, even when they are integrated along a long path, are much less than the solar photon density flux which has a temperature much higher than the saturation photon temperature. The loss caused by the molecular transition along the path is therefore much smaller than the total solar photon density, and a constant photon density is maintained, as assumed above. This justification obviously breaks down as the radiance level approaches the saturation level.

In a non-saturating region, we find that the observable photons given by Eq. (22) are controlled by the factor $N(x_o)$, the molecular density at the tangent height X. In the first-order approximation, the radiance level at a particular tangent height gives the density of the observing molecule at that altitude. For a stable molecule, its density decreases exponentially as a function of the altitude. We can model the radiance of a molecular band, as shown in Figure 7, by an exponential function with a plateau at the critical altitude. The model certainly agrees with the radiance data collected by the SPIRE for the ν_3 band of CO_2.* The computed results by the Degges program as well as by the LOWTRAN for various bands support this model.

---

*The atmospheric sounding technique currently experimented uses this model. It is to observe the saturation photon density at the critical altitude, which varies from a spectral region to another as a function of the absorption strength. The observed radiance level gives a local equilibrium temperature at the altitude. An atmospheric sounding program proposed to NESS will use the (01101-00001) and (00011-00001) transitions of CO_2.[13]

Figure 6

25

Figure 7

26
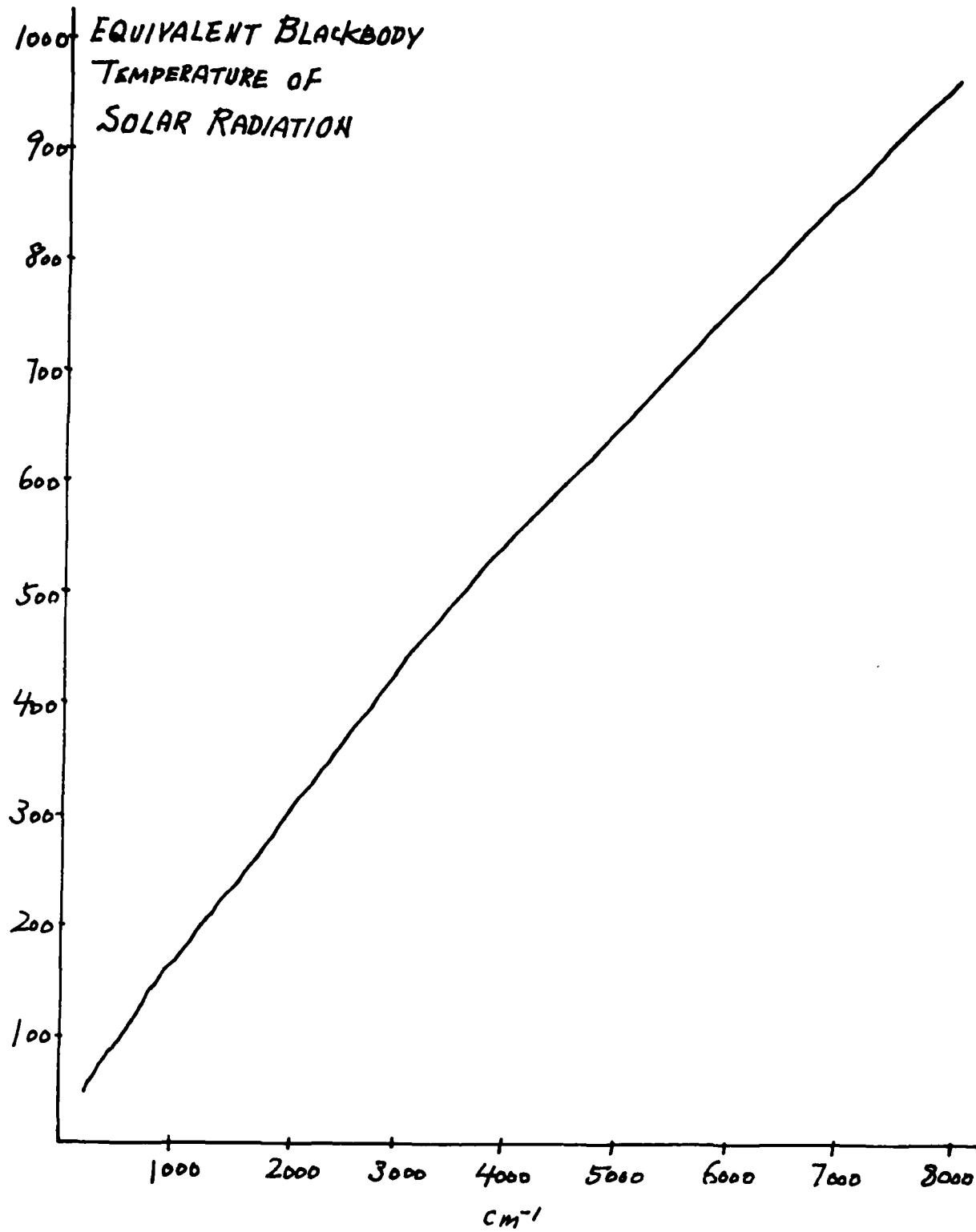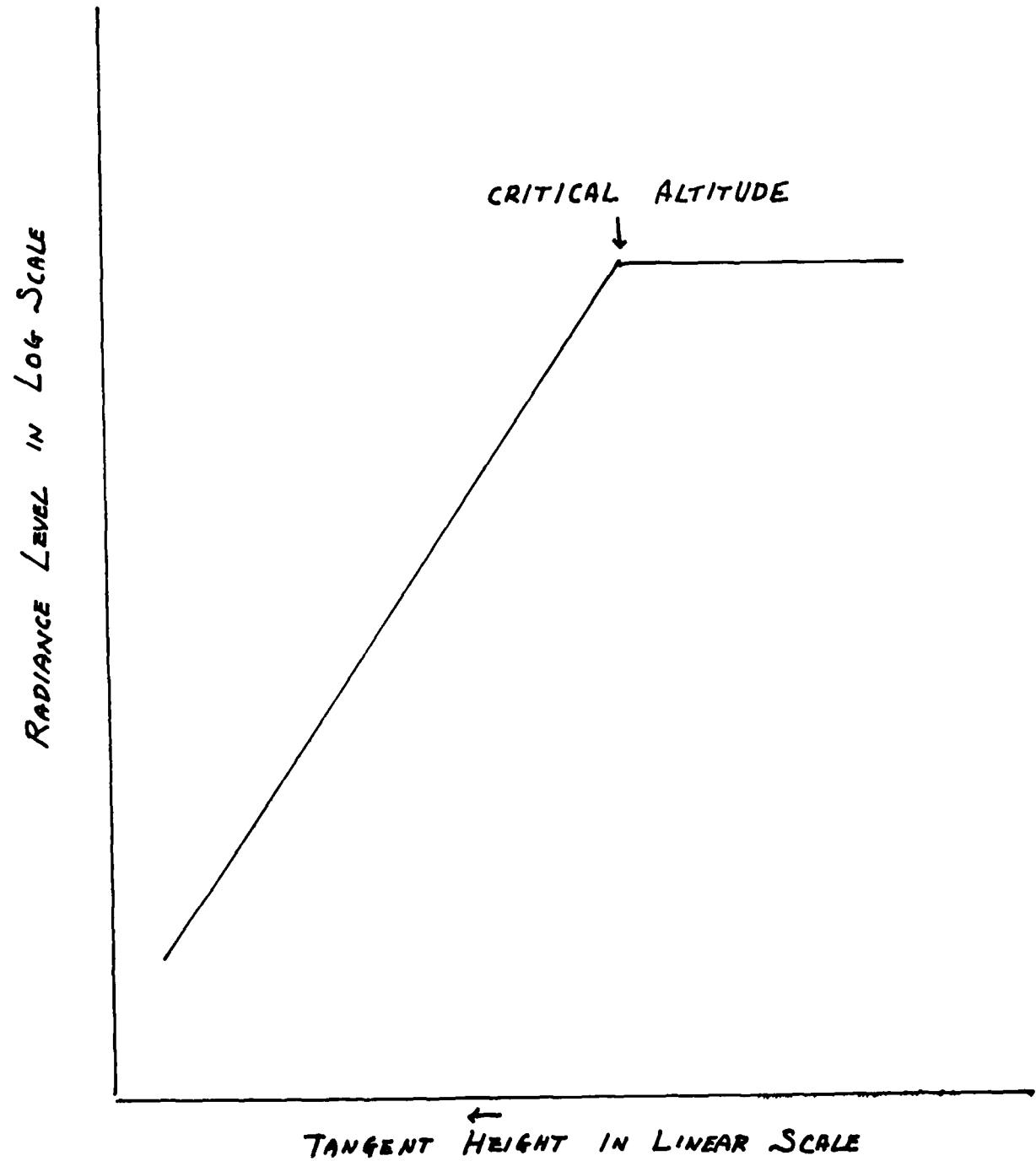
In the upper atmosphere, the radiance level is thus dictated directly by the molecular density. A deviation of the radiance level from the exponential decay is expected if the molecular concentration falls off rather differently from the exponential model. The SPIRE data on the infra-red radiance level provide a base to determine the concentration of the observed atmospheric species as a function of the altitude. From our point of view, the radiance level data of O_3, H_2O and NO and other minor species provided the first measured data which revealed the distribution of these species at high altitude.

The radiance level data of O_3 observed at a tangent height of 80 km show a rather dramatic change from the day to the night side, while the density of the excitation photons remains unchanged between these two periods. A mechanism which causes the diurnal change must be sought in some photo-chemical effect triggered by the ultraviolet photons.

Modification Applied to the BGND Program

In the course of our study, we made an extensive use of four computation programs: the BGND written by Degges of Visidyne, Inc.; the LOWTRAN by Selby[11,12] of then AFGL; the FLASH by Blättner of Radiation Research Associates, Inc.;[14] [5,6,7] and the FASCODE by Clough et al of AFGL.[15,16] We applied some modifications to these programs for improvement of the execution efficiency. The operating system of our central site computing facility is designed to accommodate a large number of the remote terminals. Because of the operating system, we found that the program executes more efficiently if its working space required is reduced in size from the original. We divided these programs into small sections, achieving their executions through our remote terminal possible during the regular hour. In addition, inefficient IO operations found in these programs are streamlined to meet our specific needs and to reduce a

27

computational waste. We have consequently achieved a substantial improvement in running these programs. Our strategy for using the FLASH program was described in our Report No. 1. Overall modifications, except for the BGND, did not extend to rewrite their computation algorithms. We feel that they have no specific merit for their coverage in this report. The case for the BGND is different. Our modification is listed in the Appendix. A general logic flow of the program execution is illustrated in Figure 8.

Other Efforts

(1) For the balloon-borne experiment, we evaluated the test data on the interferogram PCM telemetry scheme. We found some mysterious bit droppings in the PCM interferogram data string. A substantial effort was poured to resolve this mystery in a joint effort with the Idealab and the AFGL. No conclusions were made on this problem. Our effort was switched after a new contract for the balloon-borne data analysis became effective.

(2) A very efficient CRT display program for the CYBER system was written by R. Lauzzana. Its listing is given in Appendix B.

(3) In anticipating a large amount of the CYBER generated plots, we implemented a micro-computer controlled linkage between our plotting machine and the CYBER MODEM scheme. The program implemented on ROM of the 8080 system is listed in Appendix C.

Conclusion

The SPIRE experiment is in a sense a historical event in the progress of atmospheric physics. Even though the data were collected using a simple spectrometer radiometer, their merit is extremely valuable in developing our understanding of the atmospheric radiative process. Our simple-minded model

28

```
┌─────────────┐
│   BGREAD    │
└─────────────┘
       │
       ▼
┌─────────────┐
│   BANRAD    │
└─────────────┘
       │
       ▼
┌─────────────┐
│   BANINT    │
└─────────────┘


┌─────────────┐
│  Spectra 1  │
└─────────────┘
       │
       ▼
┌─────────────┐
│  Spectra 2  │
└─────────────┘


┌─────────────┐
│ Plot Program│
└─────────────┘
```

Figure 8

29
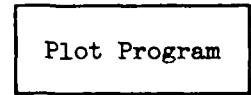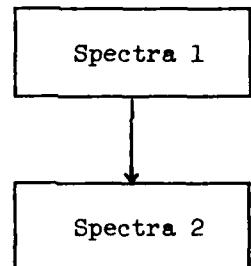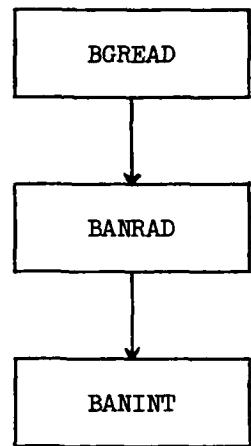
achieved a good success in understanding the radiance level of the major atmospheric CO_2 bands. We believe that our constructed model provides a good insight into the atmospheric radiative process. Ours is no different from the aforementioned computation programs, since all of these are constructed on the same theoretical development. Ours attempted to model the radiative process in the simplest form possible required for providing a good estimate of its level.

There will be no question that the data will be collected with an improved spectral resolution, as time will progress. Then we will come to understand more details of the radiative process in the atmosphere. In so doing, we will be more appreciative of those aforementioned computation programs. At this moment, our simple-minded model functioned reasonably well for analysis of the SPIRE data, revealing a basic character of the infrared radiance level in the upper atmosphere, as well as some abnormal feature which must be studied more carefully in the future.

The following list summarizes our model of the radiative process in the upper atmosphere:

(1) In the upper atmosphere, the molecular absorption transitions which trigger the emission are well approximated by assuming that they originate from the ground state.

(2) The photons available for exciting the molecular transitions are either those of the telluric origin or of the solar origin. The cross-over takes place in the vicinity of $2000 \ cm^{-1}$.

(3) Below the critical altitude, the molecular bands establish a local thermal equilibrium with the photon field. The radiative level given by the blackbody radiation formula is maintained over a large range of altitude if the absorption is very strong. It is independent

30

of the observation condition and of the scattering radiance level.

(4) Well above the critical altitude, the radiative level of the molecular band is proportional to the molecular concentration at that altitude in the first-order approximation.

(5) The scattering radiance level in the near infrared region is primarily determined by the Rayleigh process with a single scattering center.

References

1. A.T. Stair, Jr., R. Nadile, G. Frodsham, D. Baker, and W. Grieder, Atmospheric Emission Spectra (1.43-16μm) of the Earth's Limb (SPIRE), Topical Meeting on Atmospheric Spectroscopy, OSA Proc. (1978).

2. P. Hansen, R. Lauzzana, H. Sakai, and J. Strong, AFGL-TR-78-0013 (1978).

3. R.M. Goody, Atmospheric Radiation I, Oxford (1964).

4. E.J. McCartney, Optics of the Atmosphere, Wiley & Sons (1976).

5. Blattner et al, Appl. Opt. 13, 534 (1974).

6. Blattner et al, AFCRL Report TR-75-0317.

7. Blattner, RRA Research Note RRA-N7512.

8. Pauling and Wilson, Introduction to Quantum Mechanics, McGraw-Hill, New York (1935).

9. S.S. Penner, Quantitative Molecular Spectroscopy and Emissivity, Addison-Wesley (1959).

10. The transition notation for CO_2 is conformed with the AFGL Atmospheric Absorption Line Listing.

11. T. Degges, AFCRL Report TR-74-0606.

12. T. Degges and H. Smith, AFGL Report TR-77-0271.

13. A. Kaplan et al, Appl. Opt. 16, 322 (1977).

14. J. Selby and R. McClatchey, AFCRL Report TR-75-0255.

15. S.A. Clough et al, AFGL Report TR-77-0164.

16. S.A. Clough et al, AFGL Report TR-78-0081.

17. J. Kumer et al, J. Geophys. Res. 83, 4743 (1978).

```
      PROGRAM EGREAD(OUTPUT,TAPE1,TAPE2,TAPE6)
C     THIS IS THE FIRST PART CF LEGG'S BGND PROGRAM.   IT JUST READS LEGG'S
C     DATA AND GETS DATA READY FCR PROGRAM BANRAD.
      LAST CHANGE AUG 10, 1978
      DIMENSION XID(4,25),XIDT(4),ALT(30),STORE(30),IALT(25)
      DIMENSION CATATY(25),COEXT(25),COEXB(25),NLEVEL(25),NBAND(25)
      DIMENSION LSC(25,25),LCC(25,25),RCM(25,25),BV(25,25),AMASS(25)
      DIMENSION ZANG(7),STAU(7),SRAD(8),RCN(8),RATIO(8)
      DIMENSION LEC(25,25),LEL(25,25),STR(25,25),SFLUX(25,25),TFLUX(25,
     125,3)
      DIMENSION CC2(18,2),WATVA(18,2)
      DIMENSION CHEML(101)
      REAL LAMDA
      COMMON /VBRCCM/ I,CO2,WATVA,COEXT,COEXB
      COMMON /TEQCCM/ ALTE
      COMMON/KCEM/KCHEM
      COMMON/IACUT/NR,NF,NW
      COMMON CONC(201,8),T(201),LAMDA(25),
     1TCTDEN(201)
      DATA(ZANG(I),I=1,7)/0.0,20.0,40.0,60.0,70.0,80.0,90.0/
      DATA(CATATY(I),I=1,25)/8HMETHANE ,8HCARBON D,8HWATER VA,8HNITRIC C
     1,8HNITROUS ,8HOZONE PR,8HOZONE NC,18*8H        /
      KCHEM = 1
      NR = 1
      NP = 2
      NTX = 6
      NW = 6
   10 FORMAT (I2)
      READ(NR,10) IGAS
   20 FORMAT(1H1,22HNUMBER OF GAS PROFILES, I3)
      WRITE(NW,20) IGAS
      IGAS=IGAS+1
      DO 170 J=1,IGAS
   30 FORMAT (4A6,I3,F7.0)
      READ(NR,30)(XID(K,J), K=1,4), ICHK,ALTM
      IF(J.EQ.ICHK) GO TO 50
   40 FORMAT ( 7HID CARD, A6, 20HOUT OF ORDER FOR GAS, I3)
      WRITE(NW,40)(XID(K,J),K=1,4), J
   50 DO 105 K=1,6
      K1=(K-1)*5+1
      K2=K1+4
   60 FORMAT (5(F5.0,F8.1,2X),I2,I3)
      READ(NR,60)(ALT(L),STORE(L),L=K1,K2), ICHK, KCHK
      IF(ICHK.EQ.J) GO TO 80
   70 FORMAT(30HDATA CARD OUT OF ORDER FOR GAS,I3, 3X, 8HCARD NO., I3)
      IFIVE = 5
      WRITE (NW,70) J,IFIVE
   80 IF(KCHK.EQ.K) GO TO 90
      WRITE(NW,70) J,K
   90 DO 100 L=K1,K2
      IF(ALT(L).LT.ALTM) GO TO 99
      KSET=L
      GO TO 120
   99 CONTINUE
  100 CONTINUE
  105 CONTINUE
  110 FORMAT(42HMAX ALT. NOT FOUND ON DATA CARDS - GAS NO., I3)

      WRITE(NW,110) J
  120 DO 130 L=1,KSET
  130 STORE(L)=ALOG(STORE(L))
      IF(ALTM.GT.150.) GO TO 140
      IALT(J)=ALTM-59.
      GO TO 150
  140 IALT(J)=91.+(ALTM-150.)/5.
  150 CALL INTERP(ALT,STORE,KSET,1,J)
      L1=IALT(J)
      DO 160 L=1,L1
  160 CONC(L,J)=EXP(CONC(L,J))
  170 CONTINUE
      DO 175 J=1,201
  175 TCTDEN(J)=CONC(J,IGAS)
      READ(NR,30) (XIDT(K),K = 1,4), ICHK, KCHK
      IF(ICHK.EQ.1)  GO TO 190
  180 FORMAT( 7HID CARD, 4A6,26HOUT OF ORDER FOR TEMP DATA)
      WRITE(NW,180) (XIDT(K),K=1,4)
  190 DO 245 K=1,6
      K1=(K-1)*5+1
      K2=K1+4
      READ(NR,60)(ALT(L),STORE(L),L=K1,K2), ICHK, KCHK
      IF(ICHK.EQ.1) GO TO 210
  200 FORMAT(31HTEMP DATA CARD OUT OF ORDER NO.,I3)
```

33

```
          WRITE(NW,200)  K
 210 IF(KCHK.EQ.K) GO TO 230
 220 WRITE(NW,200)  K
 230 DC 240 L=K1,K2
     IF(ALT(L).LT.ALTM) GC TC 240
     KSET=L
     GC TO 260
 240 CCNTINUE
 245 CCNTINUE
 250 FCRMAT(37HMAX ALT. NCT FCUND CN TEMP DATA CARDS)
     WRITE(NW,250)
 260 CALL INTERP(ALT,SICRE,KSET,2,1)
 261 FCRMAT (1X,I5,1PSE12.4)
     WRITE(NW,261) (J,(CCNC(J,I),I=1,IGAS), T(J),J = 1,201)
     IGAS=IGAS-1
     DC 331 II=1,IGAS
     READ(NR,310)TYPE,NLEV,NBAN,AMAS,CEXT,DEXB
 310 FCRMAT (A6,22X,I2,3X,I2,3X,FE.2,4X,2(E12.4,3X))
     CC 311 J=1,25
     IF(TYPE.EC.DATATY(J)) GC TC 312
 311 CCNTINUE
     WRITE(NW,370)
 370 FCRMAT(36HTHE GAS IS NCT LISTED IN THE PROGRAM)
C    STOP 1
 312 NSF=J
     NLEVEL(NSF)=NLEV
     NBANC(NSF)=NBAN
     AMASS(NSF)=AMAS
     CCEXT(NSF)=CEXT
     CCEXB(NSF)=CEXB
     IF(NSF.EC.2)READ(NR,320) (CC2(I,1),CC2(I,2),I=1,18)
 320 FCRMAT (E(F5.2,F8.2))
     IF(NSF.EC.3)READ(NR,320) (WATVA(I,1),WATVA(I,2),I=1,18)

     ILEVL=0
     CC 313 I =1,NLEV
     ILEVL=ILEVL+1
     READ(NR,330)LSC(IJ,NSF),LCC(IJ,NSF),RCM(IJ,NSF),BV(IJ,NSF),JGAS,
    1JLEVEL
 330 FCRMAT (2I6,2E12.4,3EX,2I3)
     IF(NSF.EC.JGAS) GC TC 314
     WRITE(NW,340) NSP
 340 FCRMAT(64HTHE CODE CN THE DATA DCES NOT AGREE WITH THE FIRST CARD
    1WHICH IS,I3)
C    STOP 2
 314 IF(ILEVL.EC.JLEVEL) GC TC 313
     WRITE (NW,350) NSF
 350 FCRMAT(42HTHERE IS A CARD CUT CF SEQUENCE IN GAS NC.,I3)
C    STOP 3
 313 CCNTINUE
     ILEVL=0
     CC 316 IJ=1,NBAN
     ILEVL=ILEVL+1
     READ(NR,360)LBC(IJ,NSF),LEU(IJ,NSF),STR(IJ,NSF),SFLUX(IJ,NSF),
    1(IFLUX(I,NSF,J),J=1,3),JGAS,JLEVEL
     STR(IJ,NSF) = 2.4793E+19*STR(IJ,NSP)
 360 FCRMAT (2I6,1P5E12.4,2I3)
 362 FORMAT (2I6,1PE12.4,2I3,1X,4FE)
     IF(NSF.EC.JGAS) GC TC 322
     WRITE (NW,340) NSF
C    STOP 4
 322 IF(ILEVL.EC.JLEVEL) GC TC 316
     WRITE (NW,350) NSF
C    STOP 5
 316 CCNTINUE
 331 CCNTINUE
 378 FCRMAT (1P6E12.4)
     READ(NR,378) CHEML
A30  READ(NR,*) IGAS,NLTE
     IF(IGAS.EQ.C) STOP
     WRITE(NP,378) T
     WRITE(NP,378) TOTDEN
     WRITE(NP,378) (CCNC(NZ,IGAS),NZ=1,201)
     WRITE(NP,360) IGAS,NLTE
     NLEV = NLEVEL(IGAS)
     NBNC = NBANC(IGAS)
     WRITE(NP,335) (XIC(K,IGAS),K=1,4),NLEV,NBNC,AMASS(IGAS)
    2 ,CDEXT(IGAS),CDEXB(IGAS)
 335 FCRMAT(4A6,EX,T2,3X,I2,3X,FE.2,4X,2(E12.4,3X))
     DC 381 IJ = 1, NLEV
     WRITE (NP,360) LSC(IJ,IGAS),LCC(IJ,IGAS),RCM(IJ,IGAS),BV(IJ,IGAS)
 381 CCNTINUE
```

34

```
      CC 382 IJ = 1, NBND
      WRITE (NF,360) LFC(IJ,IGAS),LBU(IJ,IGAS),STR(IJ,IGAS)
    2 ,SFLUX(IJ,IGAS),TFLUX(IJ,IGAS,1)
382   CONTINUE
      JALT = (IALT(IGAS) - 1)/5
      WRITE (NF,360) IALT(IGAS),JALT
      IF(IGAS.NE.4) GO TC 83C
      WRITE(6,378) CHEML
      GC TO 830

      ENC

      SLBROLTINE INTERP(ALT,STCRE,KSET,MODE,I)
      CIMENSION ALT(30),STCRE(30)
      CCMMCN CCNC(201,8),T(201),LAMDA(25),
     1TCTDEN(2C1)
      J=1
      A=60.
    8 CC 120 K=1,KSET
   10 IF(A-ALT(K))80,20,20
   20 GC TC (3C,4C),MODE
   30 CCNC(J,I)=STCRE(K)
      GC TO 50
   40 T(J)=STCRE(K)
   50 IF(K.LT.KSET) GO TC 60
      RETURN
   60 IF(J.GE.91) GC TC 70
      A=A+1.
      GC TO 110
   70 A=A+5.
      GC TO 110
   80 FR=(A-ALT(K-1))/(ALT(K)-ALT(K-1))
      GC TC (9C,100), MCDE
   90 CCNC(J,I)=STCRE(K-1)+FR*(STCRE(K)-STCRE(K-1))
   91 J=J+1
      IF(J.GT.91) GC TO 92
      A=A+1.
      GC TO 10
   92 A=A+5.
      GC TC 10
  100 T(J)=STCRE(K-1)+FR*(STCRE(K)-STCRE(K-1))
      GC TO 91
  110 J=J+1
  120 CCNTINUE
      RETURN
      ENC
```

```
      PROGRAM BANRAC(TAPE2=103E,TAPE4=103B,TAPE3,OUTPUT=103E
     2 ,TAPE6=CLTPLT)
C     THIS IS A MODIFIED VERSION OF DEGG'S SUBROUTINE BANRAD
C     LAST CHANGE AUG 30, 1978
      DIMENSION FCWER(40),KL(40),XID(4)
      DIMENSION TFLUX(40),SFLUX(40)
      DIMENSION GFAC(40)
      DIMENSION LSC(40),LCC(40),LBU(40),LBC(40)
      DIMENSION WAVE(40),STR(40),BV(40),BNELAM(40),FACC(40)
      DIMENSION GNOSTE(201),TEMP(201),DENTOT(201),VIE(40,3)
      DIMENSION ENDRAD(201,40),BNETAL(201,40)
      COMMON  VIBFCP(201,25)
      COMMON /CCL/ TVSIG(201),VTSIG(201),VVLSIG(201),VVDSIG(201)
     2 ,VVNSIG(201),VVRSIG(201)
      COMMON  /CFDATA/ NALT,NLEVEL,NBAND,ZFAC,KP(40),EA(40),GUF(40)
     2 ,GDOWN(40),TTUP(40),TTDOWN(40),LPSTA(40),DOWN(40),TTP(201,40)
     3 ,CONC(201),CHEML(201)
      EQUIVALENCE (GNOSTE(1),VIBFCP(1,1))
      EQUIVALENCE (TFLUX,GUP),(SFLUX,GDOWN)
      EQUIVALENCE (TTP,BNETAL),(TTP,ENDRAD)
      EQUIVALENCE (LSC(1),VIE(1,1)),(LCC(1),VIE(1,2)),(LBU(1),VIE(1,3))
      EQUIVALENCE (GFAC,DOWN),(DENTOT(1),TTP(1,1))
      EXTERNAL CO2,H2O,FNC,C3
      ZFAC=1.0
C     READ IN TEMPERATURE AND MOLECULAR PROFILE
      READ(2,378) TEMP
      WRITE(4,378) TEMP
      READ(2,378) DENTOT
      READ(2,378) CONC
378   FORMAT(1PEE12.4)
C     READ IN WHICH GAS AND CONDITION (NLTE=3 NIGHT, NLTE=4 DAY)
      READ(2,368) IGAS,NLTE
      WRITE(4,368) IGAS,NLTE
368   FORMAT(2IE,1PEE12.4)
C     NOTE I IN DEGGS IS IGAS IN THIS PROGRAM
C     READ IN LEVEL INFO
      READ(2,310) (XID(K),K=1,4),NLEVEL,NBAND,RM,CDEXT,CDEXE
      WRITE(4,310) (XID(K),K=1,4),NLEVEL,NBAND,RM,CDEXT,CDEXE
310   FORMAT(4IE,EX,I2,3X,I2,3X,FE.2,4X,2(1PE12.4,3X))
      DO 1 I=1,NLEVEL
      READ(2,368) LSC(I),LCC(I),WAVE(I),BV(I)
      WRITE(4,368) LSC(I),LCC(I),WAVE(I),BV(I)
C     READ BAND INFO
      DO 2 I=1,NBAND
2     READ(2,368) LBC(I),LBU(I),STR(I),SFLUX(I),TFLUX(I)
      READ(2,368) NALT,XALT
      IF(IGAS.EQ.4) READ(2,50) (CHEML(I),I=1,161)
50    FORMAT(6E12.4)
C        THIS SECTION COMPUTES POPULATION OF VIBRATIONAL LEVELS.
C     COMPUTE THERMAL POPULATION
      DO 130 NZ = 1, NALT
      SUM = 0.0
      TMP = TEMP(NZ)
      DO 110 NL = 2, NLEVEL
      XYN = MOD(LSC(NL)/10,10)
      AA = EXP(-1.43879*WAVE(NL)/TMP)*XYN
      VIBFCP(NZ,NL) = AA
      SUM = SUM + AA
110   CONTINUE
      SUM = SUM + 1.0
      GNOSTE(NZ) = CONC(NZ)/SUM
      VIBFCP(NZ,NL) = CONC(NZ)*VIBFCP(NZ,NL)/SUM
120   CONTINUE
130   CONTINUE
C     FIND LEVEL COUPLED WITH NITROGEN
      DO 135 NL = 2, NLEVEL
      IF (LCC(NL) .EQ. 1)  VVWAVE = 2331.- WAVE(NL)
      IF (LCC(NL) .EQ. 2)  VVWAVE = 1556.4 - WAVE(NL)
      FAC = 2331.0
      IF (LCC(NL) .EQ. 2)  FAC = 1556.4
135   CONTINUE
      IF(NLTE.EQ.1) GO TO 170
C     FIND COLLISION EXCITATION AND DE-EXCITATION COEFS
      DO 160 NZ =1,NALT
      TEMP(NZ) = TEMP(NZ)
      TEMP13 = EXP(ALOG(TEMP(NZ))/3.0)
      IF (IGAS.EQ. 3)  GO TO 136
      IF (IGAS.NE. 2)  GO TO 140
      VTSIG(NZ) = 6.69E-10*EXP(-24.07/TEMP13)
      VVRSIG(NZ) = 1.71E-8*EXP(-175.3/TEMP13)
     1 + 6.07E-14*EXP(15.27/TEMP13)
```

```
      VVDSIG(NZ) = 1.0E-15 + 5.1EE-11*EXP(-76.75/TEMP13)
      VVNSIG(NZ) = VVRSIG(NZ)/EXP(1.43879/TEMP(NZ)*18.0)
      VVUSIG(NZ) = VVDSIG(NZ)*VIEFCF(NZ,8)/VIEPOP(NZ,5)
      GO TC 150
136   CONTINUE
      FAC = 0.C
      VVNSIG(NZ) = 4.CE-13
      VTSIG(NZ) = 5.37E-10*EXP(-70.0/TEMP13)
      VVRSIG(NZ) = VVNSIG(NZ)*EXP(-1.43879/TEMP(NZ)*1594.736)
      GO TC 150
140   CONTINUE
      VTSIG(NZ) = CDEXT
      VVNSIG(NZ) = CDEXE
      VVRSIG(NZ) = VVNSIG(NZ)*EXP(-1.43879/TEMP(NZ)*VVWAVE)
150   CCNTINUE
      TVSIG(NZ) = VTSIG(NZ)/GNDSTE(NZ)*VIEPCP(NZ,2)
      VTSIG(NZ) = VTSIG(NZ)*CENTCT(NZ)
      TVSIG(NZ) = TVSIG(NZ)*CENTCT(NZ)
      TNNV = TEMF(NZ)
      IF (ALTE .LT. 5)  GO TC 154
      IF (NZ .GT. 41 .AND. NZ .LT. 51)  TNNV = TEMP(NZ)+FLCAT(NZ-41)
     1   *280.C
      IF (NZ .GE. 51)  TNNV = 3000.0
154   CONTINUE
      FACTCR = EXP(-1.43879*FAC/TNNV)
      VVNSIG(NZ) = VVNSIG(NZ)*CENTCT(NZ)*FACTOR/(1.C + FACTCR)
      VVRSIG(NZ) = VVRSIG(NZ)*CENTCT(NZ)/(1.0 + FACTOR)
160   CCNTINUE
C     CCMPUTE VIEFCP FCR TEMF=180
      CC 166 NZ = 1, NALT
      SUM = 0.0
      TMP = 180.0
      CC 162 NL = 2, NLEVEL

      XYN = MO((LSC(NL)/10,1C)
      AA = EXF(-1.43879*WAVE(NL)/TMP)*XYN
      VIBFCP(NZ,NL) = AA
      SUM = SLM + AA
162   CCNTINUE
      SUM = SLM + 1.0
      GNDSTE(NZ) = CCNC(NZ)/SUM
      CC 164 NL = 2, NLEVEL
      VIBFCF(NZ,NL) = CCNC(NZ)*VIBFCP(NZ,NL)/SUM
164   CONTINUE
166   CCNTINUE
170   CCNTINUE
C     CCMPUTE CRTICAL DATA
      CC 180 NE = 1, NBANC
      NEC = LEC(NE)
      NI = MOC(NEC/100,100)
      NF = MOC(NEC,100)
      KL(NE) = MCC(LBU(NE)/10,10)
      KE(NE) = MCC(LEU(NE),1C)
      TWAVE = WAVE(NI) - WAVE(NF)
      ENDLAM(NE) = 1.0E+4/TWAVE
      IF (TFLLX(NE) .EQ. 0.C)  TFLLX(NE) =
     1 5.967E+6*TWAVE**4/(EXF(1.43879*TWAVE/TEMP(1)) - 1.C)
      XYM = MCC(LSC(NF)/10,1C)
      XYMM = MCC(LSC(NI)/1C,1C)
      EA(NE) = STR(NE)*2.804/ENDLAM(NE)**2*XYM
     1 *EXF(1.43879*WAVE(NF)/296.C)/XYMM
      FCWER(NE) = 1.5806EE-2C/ENDLAM(NE)
      GFAC(NE) = 3.72E-24*STR(NE)*ENDLAM(NE)**2
     1 *EXF(1.43879*WAVE(NF)/296.0)
      GUP(NE) = GFAC(NE)*TFLLX(NE)*12.4864
      IF (ALTE .LT. 3)  GUP(NE) = C.0
      GCOWN(NE) = GFAC(NE)*SFLUX(NE)
      IF (ALTE .LT. 4)  GCOWN(NE) = 0.0
      KBN = KE(NE)
      KLN = KL(NE)
      EVN=BV(1)
      ENDL = ENDLAM(NE)
      FACC(NE)= FCWER(NE)*EA(NE)
      SNB = STR(NE)*EXF(1.43879*WAVE(NF)/296.0)
      CC 180 NZ = 1, NALT
      ENDTAU(NZ,NB) = TAUMAX(KLN,KEN,RM,BVN,TEMF(NZ),ENDL,SNB,
     1 VIBFOP(NZ,NF))
      IF (ALTE .GT. 1)  GC TC 180
      ENDBAC(NZ,NB) = VIBFCF(NZ,NI)*FAC
180   CCNTINUE
      CC 175 NE=1,NBANC
175   WRITE(4,368) LBC(NE),LEL(NE),STR(NE),ENDLAM(NE),FACC(NE)
      WRITE(4,368) NALT,JALT
```

4
4

```
          IF (NLTE .EQ. 1)  GO TC 900
          ITMAX = 5
          IF(IGAS.EQ.2) ITMAX =10
          AC=.3
          ITER = 0
C         START OF BALANCE LCCF
  181 ITER = ITER + 1
          NMIN = NALT - 1

          CC 190 NB = 1, NBANC
C         LCCK CUT ENCTAU ANC ITH ARE ECUIVALENT
          TTHL=SIMF(ENDTAU(1,NB),1.0E+5,1)
          CC 182 NZ=2,NMIN
          NZM=NZ-1
          TTHNZ=SIMF(ENDTAU(NZM,NE),1.0E+5,2)
          TTH(NZM,NB)=TTHL
          TTHL=TTHNZ
  182 CCNTINUE
          TTH(NMIN,NE)=TTHL
          SUM=C.0
          IF(NLTE.LE.3) GO TC 186
          CC 184 NZ=1,NMIN
  184     SLM=SLM+TTH(NZ,NE)
  186     TTDOWN(NE)=SUM
          TTUP(NB)=C.C
          CCWN(NB)=SUM
          UFSTA(NB)=C.0
  190     CCNTINUE
  200     CCNTINUE
C         NIAL SHOLLC BE A FCWER CF TWC
          NIAL=4
          WRITE(3,1000)
 1000 FCRMAT(*  VIBFOP *)
          CC 1030 NZ=1,201
 1030     WRITE(3,8) (VIBPOF(NZ,NL),NL=1,NLEVEL)
          WRITE(6,8) (VIBPOF(12,NL),NL=1,NLEVEL)
          WRITE(6,8) (VIBPOF(32,NL),NL=1,NLEVEL)
          WRITE(3,1001)
 1001 FCRMAT(*  NALT,NBANC,ZFAC,KB(4C),EA(4() *)
          WRITE(3,*) NALT,NEANC,ZFAC
          WRITE(3,*)(KE(NB),NB=1,NEANC)
          WRITE(3,1008) (EA(ICUT),ICUT=1,16)
 1008 FCRMAT(1FEE12.4)
          GO TC (210,220,230,240,250,260,260), IGAS
  210     CONTINUE
          GO TC 350
  220     CCNTINUE
          CALL CALZ(CC2,NIAL,AC,VIB)
          GO TC 350
  230     CCNTINUE
          CALL CALZ(H2C,NIAL,AC,VIB)
          GC TC 350
  240     CCNTINUE
          CALL CALZ(FNO,NIAL,AC,VIE)
          GO TC 350
  250     CCNTINUE
          GO TC 350
  260     CONTINUE
          CALL CALZ(C3,NIAL,AC,VIE)
          GO TC 350
  350     CCNTINUE
          CC 360 NE = 1, NBANC
          NBC = LEC(NB)
          NF = MOC(NEC,100)
          KLN = KL(NE)
          KEN = KE(NE)

          EVN=BV(1)
          SNB = STR(NB)*EXF(1.43879*HAVE(NF)/296.0)
          BNCL = BACLFM(NB)
          CC 360 NZ = 1, NALT
          ENDTAU(NZ,NE) = TAUMAX(KLN,KEN,RM,EVN,TEMP(NZ),BNCL,SNB,
     1    VIBFOP(NZ,NF))
  360     CCNTINUE
  400     CCNTINUE
          IF (ITEF .LT. ITMAX)  CC TC 181
  500     CCNTINUE
          IF(NLTE .LE. 2)  GC TC 920
          CC 910 NZ = 1, NALT
          WRITE (4,8) (VIBFCP(NZ,NL), NL = 1, NLEVEL)
  910     CCNTINUE
          CC 915 NZ=1,NALT
  915     WRITE(4,8) (ENCTAL(NZ,NE),NB=1,NBAND)
  520     CCNTINUE
```

38

```
          STCP
    8     FCRMAT (1F8E12.4)
          END

C
          SLBRCLTINE CALZ(GAS,NIAL,AC,VIE)
C         THIS SUBRCLTINE CALCULATES VIE FCP FCR A GIVEN GAS AT ALL ALTITUDES
          CIMENSICA VIE(40,3)
          CCMMCN   /CFCATA/ NALT,NLEVEL,NBANC,ZFAC,KB(40),EA(40),GUE(40)
    2     ,GDCAN(40),TTUP(40),TTCCWN(40),UPSTA(40),COWN(40),TTF(201,40)
    3     ,CONC(201),CHEML(201)
          EXTERNAL GAS
          WRITE(6,*) NZ,NIAL,AC
          NZ=1
   1C     CALL CALVIE(GAS,NZ,NIAL,NEXT,AC,VIE)
          NZ=2*NIAL+NZ
          NIAL=NEXT
          IF((NZ+2*NEXT).GT.NALT) NIAL=(NALT-NZ)/2
          IF(NIAL.LE.C) RETLRN
          GC TO 10
          END

C
          SLBRCLTINE CALVIE(GAS,NZST,NIAL,NEXT,AC,VIE)
C         THIS SUBRCLTINE CALCULATES VIE FCP FCR THREE ALTICUDES AND INTERPCLA
C         BETWEEN THE END PCINTS TC CHECK ACCURACY AND CHANGES THE ALTICUE
C         INTERVAL IF NECESSARY
C         NCTE: UPSTA,DCWN,TTUF,TTCOWN MUST BE SET BEFCRE CALLING THIS PCLTINE
          CIMENSICA VIE(40,3),ATNL(1C)
          CCMMCN   VIEFCF(201,2S)
          CCMMCN /CFCATA/ NALT,NLEVEL,NBANC,ZFAC,KB(40),EA(40),GUE(40)
    2     ,GDCAN(40),TTUP(40),TTCCWN(40),UPSTA(40),COWN(40),TTF(201,40)
    3     ,CONC(201),CHEML(201)
          EXTERNAL GAS
          WRITE(3,100) NZST,NIAL,NEXT,AC
   1CC    FCRMAT(* CALVIE HERE WITH NZST=*,I5,* NIAL=*,I5,* NEXT=*,I5
    2     ,* AC=*,F5.3)
          IF(NZST.EG.1) CALL GAS(VIE(1,3),NZST)
          CC 10 NL=1,NLEVEL
   10     VIE(NL,1)=VIE(NL,3)
          CALL TTSET(NZST,NIAL,1)
          NZ=NZST+NIAL
          CALL GAS(VIE(1,2),NZ)
          CALL TTSET(NZ,NIAL,2)
          NZ=NZ+NIAL
          CALL GAS(VIE(1,3),NZ)
C         CHECK INTERF VALUES
   20     SUM=0.
          CC 30 NL=1,NLEVEL
          ATNL(NL)=ABS(ABS(VIE(NL,1)+VIE(NL,3))*.5-ABS(VIE(NL,2)))
   30     SLM=ATNL(NL)+SUM
          ATEST=SUM/(FLCAT(NLEVEL))
          WRITE(3,150) ATEST
   15C    FCRMAT(* ATEST=*,F10.3)
          IF(ATEST.LE.AC) GC TC 5C
C         REDUCE ALTITUCE RANGE IF POSSIELE
          NT=NIAL/2
          RT=FLCAT(NIAL)/2.
          IF(FLCAT(NT).NE.RT) GO TC 5C
          NIAL=NT
          CC 40 NL=1,NLEVEL
   40     VIE(NL,3)=VIE(NL,2)
          CALL TTSET(NZST,NIAL,1)
          NZ=NZST+NIAL
          CALL GAS(VIE(1,2),NZ)
          GC TO 20
C         CC QUADRATIC FIT
   50     STEP=NIAL*2
          STEP2=STEP*STEP/2.
          CC 60 NL=1,NLEVEL
          AFIT=(VIE(NL,1)+VIE(NL,3)-2.*VIE(NL,2))/STEP2
          BFIT=(VIE(NL,3)-VIE(NL,1))/STEP
          CFIT=VIE(NL,2)
          XMID=NZST+NIAL
          NZL=2*NIAL+NZST
          CC 60 NZ=NZST,NZL
          X=NZ-XMIC
C         THE VIE FCF IN VIE(I,J) IS LCG CF DENSITY
   60     VIEPOF(NZ,NL)=EXP((AFIT*X+BFIT)*X+CFIT)

C         RESET UPSTA, COWN
          CALL TTSET(NZST,2*NIAL,1)
```

39

```
      CC 70 NB=1,NEANC
      UFSTA(NB)=TTUF(NB)
70    CCWN(NB)=TTCCWN(NE)
      NEXT=NIAL
C     HAS SAFTY FACTOR AGAINS CSCILATICN CF 2.
      IF((ATEST*4.).LT.AC) NEXT=NEXT*2
      RETURN
      END

C
      SLPRCLTINE TTSET(NZST,NIAL,MCC)
C     MCC=1 TAL FROM UFSTA ANC CCWN
C     MCC=2 TAL FRCM TTLP ANC TTCCWN
      CCMMCN /CFCATA/ NALT,NLEVEL,NEANC,ZFAC,KP(40),EA(40),GUF(40)
     2 ,GDCWN(40),TTUP(4C),TTCCWN(4C),UFSTA(40),COWN(40),TTH(201,40)
     3 ,CCNC(201),CHEML(201)
      IF(MCC.NE.1) GO TC 20
      CC 10 NB=1,NBANC
      TTUP(NB)=UFSTA(NB)
10    TTDOWN(NE)=CCWN(NE)
20    NZF=NZST+NIAL-1
      CC 30 NZ=NZST,NZF
      CC 30 NB=1,NEAND
      TA=TTH(NZ,NE)
      TTUP(NB)=TTLP(NB)+TA
30    TTDOWN(NE)=TTCOWN(NE)-TA
      RETURN
      END

C
      FUNCTION RTFUN(NZ,NUS,NES)
C     NLS IS THE NUMBER CF THE LEVEL, NBS IS THE NUMEER CF THE BAND
      DIMENSICN GNCSTE(201)
      CCMMON VIBFCE(201,25)
      CCMMCN /CFCATA/ NALT,NLEVEL,NEANC,ZFAC,KP(40),EA(40),GUF(40)
     2 ,GDCWN(40),TTUP(4C),TTCCWN(4C),UFSTA(40),COWN(40),TTH(2C1,40)
     3 ,CCNC(201),CHEML(201)
      EGUIVALENCE (GNCSTE(1),VIBFCF(1,1))
      RTFUN=
     1 (XFER(VIBFOP(1,NLS),TTH(1,NES),NZ,1,1,KB(NES))
     2 + XFER(VTEFOP(NZ,NLS),TTH(NZ,NES),NALT,NZ,2,KB(NES)))*EA(NBS)
     3 +GDCWN(NES)*SM1(KP(NES),ZFAC*TTOCWN(NBS))
     4 +GUF(NES)*SN12(KE(NES),TTUF(NPS))
      RETURN
      END

      SLPRCLTINE CC2(VIFI,NZ)
      DIMENSICN VIEI(40),GNCSTE(201)
      CCMMCN /CGL/ TVSIG(201),VTSIG(201),VVUSIG(201),VVDSIG(201)
     2 ,VVNSIG(201),VVRSIG(2C1)
      CCMMCN VIEFCF(201,25)
      CCMMCN /CFCATA/ NALT,NLEVEL,NEANC,ZFAC,KP(40),EA(40),GUF(4C)
     2 ,GDCWN(40),TTUP(4C),TTCCWN(4C),UFSTA(40),COWN(40),TTH(2C1,40)
     3 ,CCNC(2C1),CHEML(201)
      EGUIVALENCE (GNCSTE(1),VIEFCF(1,1))
      VIBI(8) =ALCG( GNCSTE(NZ)*(VVNSIG(NZ) + RTFUN(NZ,P,S))
     2 /(VVRSIG(NZ) + EA(9) + VVCSIG(NZ)))
      TVS = TVSIG(NZ)
      VTS = VTSIG(NZ)
      FOP3 = VIEFCP(NZ,3)
      FOP4 = VIEFCP(NZ,4)
      FCP7 = VIEFOP(NZ,7)
      FOP2 = ((TVS + RTFUN(NZ,2,1))*GNCSTE(NZ)
     2 + FCP3*(EA(2) + VTS) + FOP4*(EA(3) + VTS)
     3 + FCP7*(EA(7) + VTS))/(VTS + EA(1) + TVS)
      VIBI(2) =ALOG(POF2)
      FOP8 = VIPFCF(NZ,8)
      FCP7 = ((TVS*0.25 + GLF(7)*SN12(KP(7),TTUP(7))
     1 + GCOWN(7)*SM1(KE(7),TTCOWN(7))*FCP2
     2 + EA(8)*FCP8 + EA(15)*VIBFCF(NZ,9) + EA(16)*VIBFCP(NZ,10))
     3 /(EA(7) + VTS)
      VIEI(7)=ALCG(POP7)
      FCP5 = VIEFCP(NZ,5)
      FCP6 = VIBFCP(NZ,6)
      FOP3 = (FCF2*(0.25*TVS + GUP(2)*SN12(KP(2),TTLP(2))
     1 + GCOWN(2)*SM1(KE(2),TTCOWN(2)) + FCF5*(EA(4) + VTS)
     2 + EA(10)*FCFR + EA(13)*VIEFCF(NZ,9) +EA(14)*VIBFCP(NZ,10))
     3 /(EA(2) + VTS + TVS)
      VIEI(3)=ALCG(FOP3)
      FOP4 = (FCF2*(0.E*TVS + GUP(3)*SN12(KE(3),TTUF(3))
     1 + GCOWN(3)*SM1(KE(3),TTCOWN(3))) + FCP5*EA(5)
     2 + FCP6*(EA(6) + VTS))/(EA(3) + VTS + TVS)
```

40

```
      VIEI(4)=ALCG(POP4)
      POP5 = (FCF3*(TVS + GLP(4)*SN12(KE(4),TTUF(4))
     1 + GDOWN(4)*SM1(KE(4),TTCOWN(4)))
     2 + FCP4*(GUP(5)*SN12(KE(5),TTUF(5))
     3 + GDOWN(5)*SM1(KE(5),TTCOWN(5)) + POP8*VVDSIG(NZ))
     4 /(EA(4) + EA(5) + VTS + VVUSIG(NZ))
      VIEI(5)=ALCG(POP5)
      FCP6 = (FCF4*(0.5*TVS + GUP(6)*SN12(KE(6),TTUF(6))
     1 + GDOWN(6)*SM1(KE(6),TTCOWN(6))))/(EA(6) + VTS)
      VIEI(6)=ALCG(FOP6)
      VIBI(9)=ALCG((VIEFOF(NZ,3)*(VVNSIG(NZ) + RTFLN(NZ,9,13))
     1 + GNDSTE(NZ)*RTFUN(NZ,9,11) + VIBPCP(NZ,7)*RTFUN(NZ,9,15))
     2 /(VVRSIG(NZ) + EA(11) + EA(13) + EA(15))
      VIBI(10)=ALCG((VIBFCP(NZ,7)*(VVNSIG(NZ) + RTFLN(NZ,10,16))
     1 + VIBFCF(NZ,3)*RTFUN(NZ,10,14) + GNDSTE(NZ)*RTFUN(NZ,10,12))
     2 /(VVRSIG(NZ) + EA(12) + EA(14) + EA(16)))
      VIBI(1) = CONC(NZ) - (FOP8 +FOP6 + POP5 + POP7 + POP4
     1 + PCP3 + FCF2)
      IF (VIBI(1) .LE. 0.0)  VIPI(1) = 0.05*CONC(NZ)

      VIPI(1)=ALCG(VIBI(1))
      RETURN
      END

C

      SUBROUTINE H2C(VTEI,NZ)
      DIMENSION VIBI(40),GNDSTE(201)
      COMMON  VIEPCP(201,25)
      COMMON /CCL/ TVSIG(201),VTSIG(201),VVUSIG(201),VVDSIG(201)
     2 ,VVNSIG(201),VVRSIG(201)
      COMMON /CFDATA/ NALT,NLEVEL,NBAND,ZFAC,KB(40),EA(40),GUF(40)
     2 ,GDOWN(40),TTUP(40),TTCOWN(40),UPSTA(40),COWN(40),TTH(201,40)
     3 ,CONC(201),CHEML(201)
      EQUIVALENCE (GNDSTE(1),VIBFCF(1,1))
      VIBI(2)=ALCG( ((TVSIG(NZ) +VVRSIG(NZ)
     1 + RTFUN(NZ,2,1))*GNDSTE(NZ) +VIEFCF(NZ,3)*EA(5))
     2 /(VTSIG(NZ) + EA(1) + VVNSIG(NZ))
      VIBI(3)=ALCG(((TVSIG(NZ) + VVRSIG(NZ)
     1 + RTFLN(NZ,3,5))*VIBFCF(NZ,2) +
     2 RTFLN(NZ,3,2)*GNDSTE(NZ) + VIBFCP(NZ,6)*EA(11))
     3 /(EA(2) + EA(5) + VTSIG(NZ) + VVNSIG(NZ)))
      VIBI(4)=ALCG((RTFUN(NZ,4,3)*GNDSTE(NZ)
     1 + RTFUN(NZ,4,6)*GNDSTE(NZ))/(EA(3) + EA(6)))
      VIBI(5)=ALCG((RTFUN(NZ,5,4)*GNDSTE(NZ)
     1 + RTFLN(NZ,5,7)*VIBFCP(NZ,2))/(EA(4) + EA(7)))
      VIBI(6)=ALCG((RTFUN(NZ,6,8)*VIBFCF(NZ,8)
     1 + RTFUN(NZ,6,12)*GNDSTE(NZ))/(EA(8) + EA(11)
     2 + EA(12)))
      VIBI(7)=ALCG((RTFLN(NZ,7,9)*VIBFOP(NZ,2)
     1 + RTFUN(NZ,7,13)*GNDSTE(NZ))/(EA(9) + EA(13)))
      VIBI(8)=ALCG((RTFUN(NZ,8,10)*VIEFOP(NZ,2)
     1 + RTFUN(NZ,8,14)*GNDSTE(NZ))/(EA(10) + EA(14)))
      VIEI(1)=ALCG(CONC(NZ)-(VIEFCF(NZ,8)+VIBPCP(NZ,7)+VIBFCF(NZ,6)
     2 +VIBFOP(NZ,5)+VIEFCF(NZ,4)+VIBFCF(NZ,3)+VIBFCF(NZ,2)))
      RETURN
      END

      SUBROUTINE RNC(VIEI,NZ)
      DIMENSION VIBI(40),GNDSTE(201)
      COMMON  VIBFCF(201,25)
      COMMON /CCL/ TVSIG(201),VTSIG(201),VVUSIG(201),VVDSIG(201)
     2 ,VVNSIG(201),VVRSIG(201)
      COMMON /CFDATA/ NALT,NLEVEL,NBAND,ZFAC,KB(40),EA(40),GUF(40)
     2 ,GDOWN(40),TTUP(40),TTCOWN(40),UPSTA(40),COWN(40),TTH(201,40)
     3 ,CONC(201),CHEML(201)
      EQUIVALENCE (GNDSTE(1),VIEPCF(1,1))
      VIBI(2)=ALCG(((TVSIG(NZ) + VVNSIG(NZ)
     1 + GLP(1)*SN12(KE(1),TTUF(1))
     2 + (XFER(VIBPOP(1,2),TTH(1,1),NZ,1,1,KB(1))
     3 + XFER(VIEPCP(NZ,2),TTH(NZ,1),NALT,NZ,2,KB(1)))*EA(1)
     4 + GDOWN(1)*SM1(KE(1),TTCOWN(1)))*GNDSTE(NZ)
     5 + (EA(3) + VTSIG(NZ))*VIBFCF(NZ,3) + 2.0*CHEML(NZ))
     6 /(EA(1) + VTSIG(NZ) + VVRSIG(NZ) + TVSIG(NZ)))

      VIBI(3)=ALCG(((TVSIG(NZ)
     1 + GLP(3)*SN12(KE(3),TTUF(3))
     2 + GLP(2)*SN12(KE(2),TTUF(2))
     3 + GDOWN(2)*SN12(KE(2),TTCOWN(2))
     4 + GDOWN(3)*SN12(KE(3),TTCOWN(3)))*VIEPCP(NZ,2)
     5 + 2.0*CHEML(NZ))/(EA(2) + EA(3) + TVSIG(NZ)))

      VIBI(1)=ALCG(CONC(NZ) - VIEFCF(NZ,2) - VIBPCP(NZ,3))
```

```fortran
u         RETURN
          END

C
          SUBROUTINE C3(VIBI,NZ)
          DIMENSION VIBI(40),GNDSTE(201)
          COMMON   VIBPOP(201,25)
          COMMON /CCL/ TVSIG(201),VTSIG(201),VVUSIG(201),VVDSIG(201)
         2 ,VVNSIG(201),VVRSIG(201)
          COMMON   /CFDATA/ NALT,NLEVEL,NBAND,ZFAC,KB(40),EA(40),GUP(40)
         2 ,GDOWN(40),TTUP(40),TTDOWN(40),UPSTA(40),DOWN(40),TTH(201,40)
         3 ,CONC(201),CHEML(201)
          EQUIVALENCE (GNDSTE(1),VIBPOP(1,1))
          VIBI(2)=ALOG((TVSIG(NZ) + VVNSIG(NZ)                         008
         1   + GUP(1)*SN12(KB(1),TTUP(1))                                E5
         2   + (XFER(VIBPOP(1,2),TTH(1,1),NZ,1,1,KB(1))                  I
         3   + XFER(VIBPOP(NZ,2),TTH(NZ,1),NALT,NZ,2,KB(1)))*EA(1)       I
         4   + GDOWN(1)*SN1(KB(1),TTDOWN(1)))*GNDSTE(NZ)                 EE
         5   /(VTSIG(NZ) + EA(1) + VVRSIG(NZ)))
n
          VIBI(1)=ALOG(CONC(NZ)-VIBPOP(NZ,2))
          RETURN
          END

C
          FUNCTION XFER (RAD,TTH,NTOP,NBOT,K,KB)
          COMMON  VIBPOP(201,25)
          DIMENSION RAD(161),TTH(161)
          NINT = NTOP - NBOT
          NPL = NINT + 1
          IF (NINT .GT. 0)  GO TO 100
          XFER = 0.0
          RETURN
  100     CONTINUE
          RA = 1.0
          TB = 0.0
          EA3 = 0.0
          IF (K .GT. 1)  GO TO 200
          NF = NTOP
          RFAC = VIBPOP(NF,1)/RAD(NF)
          SUM = 0.5*RA
          DO 120 NZ = 1, NINT
          NY = NINT - NZ + 1
          TM = TTH(NY)
          TB = TB + TM
          EB3 = SLO3(KB,TB)
          NF = NF - 1
          RB = RFAC*RAD(NY)/VIBPOP(NF,1)
          SUM = SUM + (RB - RA)*(EB3 - EA3)/TM
          RA = RB
          EA3 = EB3
  120     CONTINUE
          XFER = ABS(SUM - RB*SN12(KB,TB))/RFAC
          RETURN
  200     CONTINUE
          NF = NBOT
          RFAC = VIBPOP(NF,1)/RAD(1)
          SUM = 0.5*RA
          DO 220 NZ = 2, NPL
          TM = TTH(NZ - 1)
          TB = TB + TM
          EB3 = SLO3(KB,TB)
          NF = NF + 1
          RB = RFAC*RAD(NZ)/VIBPOP(NF,1)
          SUM = SUM + (RB - RA)*(EB3 - EA3)/TM
          RA = RB
          EA3 = EB3
  220     CONTINUE
          XFER = ABS(SUM - RB*SN12(KB,TB))/RFAC
          RETURN
          END

          FUNCTION SLO3(K,TAU)
C
C         COMPUTE ONE-HALF THE VALUE OF THE FUNCTION L SUB NOUGHT THREE.
C         K .EQ. 1, PARALLEL BAND
C         K .EQ. 2, PERPENDICULAR BAND

          T = ABS(TAU)
          IF (T .EQ. 0.0)  GO TO 300
          IF (K .GE. 2)  GO TO 200
```

```
      IF (T .GT. 1.0)  GC TC 110
      X = (T - 0.5)*2.C
      SL03 = (((2.583E-4*X - 6.0145E-3)*X - 0.102723))*X
     1 + 0.12530021*ALCG(T)*T + (.4035503)*T
      RETURN
110   CONTINUE
      IF (T .GT. 4.0)  GC TC 120
      X = (T - 2.5)/1.5
      SL03 = (((((7.666E-4 - 3.516E-4*X)*X - 1.0464E-3)*X
     1 + 2.7618E-3)*X - 7.9E629E-3)*X + 0.0219502)*X
     2 - C.0626550)*X + 0.1575702)*T
      RETURN
120   CCNTINUE
      IF (T .GT. 10.0)  GC TC 13C
      X = (T - 7.0)/3.C
      SL03 = ((((((1.7455E-4*X - 4.9951E-4)*X + 0.00113772)*X
     1 - 0.00334538)*X + 0.010098E1)*X - 0.03090774)*X
     2 + C.1086302)*T
      RETURN
130   CCNTINUE
      Y = ALCG(T)
      IF (T .GT. 100.0)  GO TC 140
      X = 9.21034037197E/Y - 3.C
      SL03 = ((((((8.93E-5 - 5.78E-5*X)*X + 2.458E-4)*X
     1 - 0.0012858)*X + 0.0340297)*X + 0.3210431)*Y
      RETURN
140   CCNTINUE
      IF (T .GT. 1000.C)  GC TC 150
      X = 27.631021115C3/Y - 5.C
      SL03 = (((((8.3E-6 - 2.01E-5*X)*X - 1.843E-4)*X
     1 + 0.0125309)*X + 0.2729344)*Y
      RETURN
150   CONTINUE
      X = 13.8155105575E/Y - 1.C
      SL03 = ((((((5.366E-4*X - 0.0015038)*X + 0.0016309)*X
     1 - 0.0001371)*X - 0.0023757)*X + 0.0301341)*X
     2 + C.2915830)*Y
      RETURN
200   CCNTINUE
      IF (T .GT.1.0)  GC TC 210
      X = (T - 0.5)*2.0
      SL03 = (((1.607E-4*X - 4.0565E-3)*X - 0.0880205)*X
     1 + C.097407381*ALCG(T)*T + 0.4161293)*T
      RETURN
210   CCNTINUE
      IF (T .GT. 4.C)  GC TC 220
      X = (T - 2.5)/1.5
      SL03 = ((((((5.904E-4 - 2.794E-4*X)*X - 8.06SE-4)*X

     1 + 2.2244E-3)*X - 6.6604E-3)*X + 0.0197235)*X
     2 - C.0631562)*X + 0.2307133)*T
      RETURN
220   CONTINUE
      IF (T .GT. 10.0)  GC TC 230
      X = (T - 7.0)/3.C
      SL03 = ((((((1.476E-4*X - 4.295E-4)*X + 1.0472E-3)*X
     1 - 3.316E-3)*X + 0.0107584)*X - 0.0356259)*X
     2 + C.1310340)*T
      RETURN
230   CCNTINUE
      Y = ALCG(T)
      IF (T .GT. 100.0)  GC TC 240
      X = 9.21034037197E/Y - 3.0
      SL03 = (((((4.166E-4 - (2.131E-4*X + 3.79E-5)*X)*X
     1 + 7.79E-5)*X + 5.84E-5)*X - 0.0022069)*X
     2 + 0.0330089)*X + 0.4187794)*Y
      RETURN
240   CCNTINUE
      IF (T .GT. 1000.C)  GC TC 25C
      X = 27.631021115C3/Y - 5.0
      SL03 = (((1.66E-5*X - 3.642E-4)*X + 0.0134767)*X
     1 + 0.3700589)*Y
      RETURN
250   CCNTINUE
      X = 13.8155105575E/Y - 1.0
      SL03 = ((((((1.0049E-3 - 4.412E-4*X)*X - 8.441E-4)*X
     1 + 9.451E-4)*X - 3.476E-3)*X + 0.0335987)*X
     2 + C.3254546)*Y
      RETURN
300   CONTINUE
      SL03 = 0.0
      RETURN
      END
```

```
      FUNCTION SN12 (K,TAU)
      T = ABS(TAU)
      IF (T .EG. 0.0)  GC TC 300
      IF (K .IT. 1)  GC TC 200
      IF (T .GT. 1.0) GC TC 110
      T = 25.2676037*T
      SN12 = ((((((5.54E-16*T - 1.3747E-13)*T + 3.11917E-11)*T
     1  - 6.43600E-9)*T + 1.24752E-6)*T - 2.57626E-4)*T
     2  - 0.0826424E + 0.01984964*ALOG(T))*T + 1.0)*0.5
      RETURN
  110 CONTINUE
      IF (T .GT. 4.0)  GC TC 120
      X = (T - 2.5)/1.5
      SN12 = ((((((3.00561E-3 - 1.41466E-3*X)*X - 3.98085E-3)*X
     1  + 9.48736E-3)*X - 0.02794C7)*X + 0.0523088)*X
     2  - 0.1045984)*X + 0.18627431)*0.5
      RETURN
  120 CONTINUE
      IF (T .GT. 10.0)  GC TC 130
      X = (T - 7.0)/3.0
      SN12 = (((((5.08470E-4 - 1.90181E-4*X)*X - 5.27594E-4)*X
     1  + 2.24274E-3)*X - 6.542685E-3)*X + 0.01290804)*X
     2  - 0.02597975)*X + 0.06552257)*0.5
      RETURN
  130 CONTINUE
      IF (T.GT. 100.0)  GC TC 140
      Y = ALOG(T)
      X = 9.21034037197E/Y - 3.0
      SN12 = ((((( 7.836E-4*X - 1.5181E-3)*X + 9.0128E-4)*X
     1  - 6.496E-4)*X - 1.0632E-3)*X + 0.0149364)*X
     2  + 0.43782273)*0.5/T
      RETURN
  140 CONTINUE
      Y = ALOG(T)
      X = 9.21034037197E/Y - 1.0
      SN12 = ((((5.074EE-3 - 4.681E-4*X)*X - 1.5848E-3)*X
     1  + 0.01475267)*X + 0.4079074)*0.5/T
      RETURN
  200 CONTINUE
      IF (T .GT. 1.0)  GC TC 210
      T = 26.0729185*T
      SN12 = ((((((2.71E-16*T - 6.9372E-14)*T + 1.58314E-11)*T
     1  - 3.39873E-9)*T + 7.01818E-7)*T - 1.61594E-4)*T
     2  - 0.0656453 + 0.01494384*ALOG(T))*T + 1.0)*0.5
      RETURN
  210 CONTINUE
      IF (T .GT. 4.0)  GC TC 220
      X = (T - 2.5)/1.5
      SN12 = ((((((2.32E-3 - 1.1203E-3*X)*X - 3.0489E-3)*X
     1  + 7.934EE-3)*X - 0.0213281)*X + 0.0521586)*X
     2  - 0.1213660)*X + 0.2508902)*0.5
      RETURN
  220 CONTINUE
      IF (T .GT. 10.0)  GC TC 230
      X = (T - 7.0)/3.0
      SN12 = ((((((4.727E-4*X - 1.2459E-3)*X + 2.5263E-3)*X
     1  - 6.7323E-3)*X + 0.0174769)*X - 0.0421443)*X
     2  + 0.0958415)*0.5
      RETURN
  230 CONTINUE
      IF (T .GT. 100.0)  GC TC 240
      Y = ALOG(T)
      X = 9.21034037197E/Y - 3.0
      SN12 = ((((9.1E-5*X - 1.605EE-3)*X - 1.8059E-3)*X
     1  + 0.0263129)*X + 0.6392594)*0.5/T
      RETURN
  240 CONTINUE
      Y = ALOG(T)
      X = 9.21034037976/Y - 1.0
      SN12 = ((((6.444E-4*X + 3.463E-4)*X - 5.059E-4)*X
     1  + 0.0216100)*X + 0.6910521)*0.5/T
      RETURN
  300 CONTINUE
      SN12 = 0.5
      RETURN
      END
```

```fortran
      FUNCTION TAUMAX (KL,KE,RM,BV,T,FLAM,S,DEN)
      SIG = 1.4387C*BV/T
      TAUMAX = 5.3E13E-10*SCRT(RM/T)*FLAM*DEN*S*SJMAX(KL,KE,SIG)
      RETURN
      END


      FUNCTION SJMAX (KL,KE,SIG)
C     SJMAX COMPUTES THE RELATIVE STRENGTH OF THE STRONGEST LINE
C       IF KB = 1, SIGMA-SIGMA TRANSITION
C       IF KB = 2, SIGMA-PI TRANSITION
      GO TC (110,120,130), KE
  110 CONTINUE
      TJM = 0.5*(SCRT(0.25 + 2.0/SIG) - 1.5)
      A = C.5*(TJM + TJM + 1.0)
      GO TC 200
  120 CONTINUE
      TJM = 0.5*(SCRT(2.0/SIG) - 1.0)
      A = TJM + 1.0
      GO TC 200
  130 CONTINUE
      SJMAX = 0.0
      RETURN
  200 CONTINUE
      SJM = A*EXF(-SIG*TJM*(TJM + 1.0))
      SJMAX = SJM/CJPART(KL,SIG)
      RETURN
      END

      FUNCTION CJPART (K,SIG)
C     CJPART COMPUTES THE ROTATIONAL PARTITION FUNCTION FOR A
C       RIGID ROTATOR, SIG .LE. 0.2.
C       K = 1, ALL   LEVELS POPULATED
C       K = 2, EVEN LEVELS POPULATED
C       K = 3, CDD  LEVELS POPULATED
      IF (SIG .GT. 0.2)  STOP
      X = SIG
      IF (K - 2)  110,120,130
C     COEFFICIENTS ARE FROM RUN CALTE-EV
  110 CONTINUE
      CJPART = ((((0.00386500232857*SIG + 0.01257180035471)*X
     1 + 0.066675616453C)*X + 0.33333133337S)*X + 1.00000000017)/SIG
      RETURN
  120 CONTINUE
      CJPART = ((((((2.5079214103*SIG - 29.5697177693)*SIG
     1 + 5.4952670096)*SIG - 0.505072235E08)*SIG
     2 + 0.025737819354E)*SIG + 0.032835050950E)*SIG
     3 + 0.16667022077D)*SIG + 0.49999999456)/SIG
      RETURN
  130 CONTINUE
      CJPART = (((((((29.570311787E - 62.507459E502*SIG)*SIG
     1 - 5.4957937366S)*SIG + 0.50575509427C1)*SIG
     2 - 0.01103975167E)*SIG + 0.0333152850E09)*SIG
     3 + 0.16666311238D7)*SIG + .500000000044)/SIG
      RETURN
      END

      FUNCTION SIMP (A,H,K)
      DIMENSICN A(3)
      IF (K - 2)  100,110,120
  100 CONTINUE
      SIMF = (8.0*A(2) + 5.0*A(1) - A(3))*H/12.0
      RETURN
  110 CONTINUE
      SIMF = (8.0*A(2) + 5.0*A(3) - A(1))*H/12.0
      RETURN
  120 CONTINUE
      SIMP = (4.0*A(2) + A(1) + A(3))*H/3.0
      RETURN
      END
```

```fortran
      FUNCTION SM1 (K,TAU)
      T = ABS(TAU)
      IF (T .EQ. 0.0)  GC TC 300
      IF (K .GE. 2)  GC TC 200
      IF (T .GT. 4.0)  GO TC 110
      X = (T - 2.0)*0.5
      SM1 = (((((2.35450E-03*X - 8.90535E-03)*X + 2.56223E-02)*X
     1  - 7.07877E-02)*X + 1.63735E-01)*X - 3.01749E-01)*X + 0.426845
      RETURN
110   CONTINUE
      IF (T .GT. 10.0)  GC TC 120
      X = (T - 7.0)/3.0
      SM1 = (((((5.92257E-04*X - 1.79043E-03)*X + 4.05109E-03)*X
     1  - 1.04749E-02)*X + 2.59109E-02)*X - 6.01706E-02)*X + 0.134123
      RETURN
120   CCNTINUE
      IF (T .GT. 100.0)  GC TC 130
      Y = ALOG(T)
      X = 9.210340371976/Y - 3.0
      SM1 = ((((((-6.24516E-04*X - 6.79378E-04)*X + 2.98590E-03)*X
     1  -1.28150E-03)*X - 6.31784E-03)*X + 3.78904E-02)*X
     2  + 8.90441E-01)/T
      RETURN
130   CONTINUE
      Y = ALOG(T)
      X = 9.210340371976/Y - 1.0
      SM1 = ((((((6.12038E-04*X - 2.93434E-04)*X - 4.36163E-04)*X
     1  + 1.30037E-02)*X - 3.80088E-04)*X + 3.23441E-02)*X
     2  + 8.17407E-01)/T
      RETURN
200   CCNTINUE
      IF (T .GT. 4.0)  GC TC 210
      X = (T - 2.0)*0.5
      SM1 = (((((1.39071E-03 *X - 5.29904E-03)*X + 1.58249E-02)*X
     1  - 4.74039E-02)*X + 1.26175E-01)*X - 2.86333E-01)*X + 0.517573
      RETURN
210   CCNTINUE
      IF (T .GT. 10.0)  GC TC 220
      X = (T - 7.0)/3.0
      SM1 = (((((3.76420E-04*X - 1.25384E-03)*X + 3.41168E-03)*X
     1  - 1.05360E-02)*X + 3.07476E-02)*X - 8.14705E-02)*X + 0.1941308
      RETURN
220   CONTINUE
      IF (T .GT. 100.0)  GO TC 230
      Y = ALOG(T)
      X = 9.210340371976/Y - 3.0
      SM1 = (((-7.53415E-03*X - 8.07124E-03)*X + 6.62199E-02)*X
     1  + 1.304245)/T
      RETURN
230   CONTINUE
      Y = ALOG(T)
      X = 9.210340371976/Y - 1.0
      SM1 =((((( 1.09067E-03 - 5.82510E-04*X)*X + 1.99669E-03)*X
     1  + 1.21174E-03)*X + 4.76166E-02)*X + 1.18445)/T
      RETURN
300   CCNTINUE
      SM1 = 1.0

      RETURN
      END
```

```
        FROGRAM EANINT (CUTPUT,TAPE3,TAPE4,TAPE6)
C       THIS FROGRAM IS LIKE THE LAST PART OF CEGG'S ECNC
        LAST CHANGE AUG 22, 1977
        CIMENSION XIC(4),FAC(25),RLAM(25)
        CIMENSICN LSC(25),LCC(25),RCM(25),BV(25)
        CIMENSION ZANG(7),STAU(7),SRAC(8)
        CIMENSION LEC(25),LBU(25),STR(25)
        CCMMCN /ECRCOM/ VIBPCF(201,10)
        CCMMCN TAU(201,25),RAC(201),T(201)
        CATA(ZANG(I),I=1,7)/0.0,20.0,40.0,60.0,70.0,80.0,90.0/
  360 FCRMAT (2IE,1F5E12.4,2I3)
  362   FCRMAT (2IE,1PE12.4,2I3,1X,4AE)
        READ(4,378) T
        READ(4,362) IGAS,NLTE
        READ(4,310) (XIC(K),K=1,4),NLEVEL,NBANC,AMASS
  310 FCRMAT(4AE,EX,I2,3X,I2,3X,FE.2)
   37 ( FCRMAT (1F6E12.4)
        WRITE (3,360) IGAS
        WRITE (3,362) NLEVEL,NEANC,AMASS,IGAS,NLTE,(XIC(K),K=1,4)
        CC 381 IJ = 1, NLEVEL
        READ(4,360) LSC(I),LCC(IJ),RCM(IJ),BV(IJ)
        WRITE (3,360) LSC(IJ),LCC(IJ),RCM(IJ),BV(IJ)
  381   CCNTINUE
        CC 382 I  = 1, NEANC
        READ(4,360) LEC(IJ),LBU(IJ),STR(IJ),RLAM(IJ),FAC(IJ)
        WRITE (3,360) LEC(IJ),LBU(IJ),STR(IJ)
  382   CCNTINUE
        READ(4,360) IALT,JALT
        WRITE (3,360) IALT,JALT
    8 FCRMAT(1F8E10.4)
  110   FCRMAT(1F8E12.4)
        CC 331 NZ=1,IALT
  331   READ(4,110) (VIBPCF(NZ,NL),NL=1,NLEVEL)
        CC 335 NZ=1,IALT
  335 WRITE(3,8) (VIBPCF(NZ,NL),NL=1,NLEVEL)
        CC 337 NZ=1,IALT
  337   READ(4,110) (TAU(NZ,NB),NB=1,NEANC)
        JSET=NEANC
        CC 840 J=1,JSET
        NEC=LEC(J)
        NI=MOC(NEC/100,100)
        CC 600 NZ=1,IALT
  600   RAC(NZ)=FAC(J)*VIBPCF(NZ,NI)
  700 FCRMAT(1H1,4AE, 8HEANC NC.,I3,EX,7HLAMDA =,F6.2,7HMICRC M,4X,4AE//)
        WRITE(6,700)(XIC(K),K=1,4),J,RLAM(J),(XIC(K),K=1,4)
        IF (NLTE .GT. 1)  GC TC 707
  704 FCRMAT(4(H ASSUMES LCCAL THERMCCYNAMIC EQUILIERIUM/)
  705 WRITE(6,704)
        GC TO 715
  706   FCRMAT (28H LACK CF THERMAL EQUILIBRIUM ,I6/)
  707 WRITE (6,706) NLTE
  710 FCRMAT(118H EXOATMCSPHERIC VIEWING $$ - - - - - - - - - - - - E N C
       1 C A T M C S P H E R I C   V I E W I N G - - - - - - - - - - - )
  715 WRITE (6,710)
  720 FCRMAT(118H TAN HT  LINE BANC RAC  $$  ALTC - - - - - - BAND RADI
       1ANCE AS A FUNCTION CF ZENITH ANGLE (WATTS/CM2-STER)- - - - - )
        WRITE (6,720)
  730 FCRMAT(118H (KM) (WATTS/CM2-STER) $$   (KM)   C DEG - UE - 20 CEG
       1    40 CEG        EC CEG       7C CEG       80 DEC   90 CEG-HCRZ )
        WRITE (6,730)
  740   FCRMAT (FE.0,1PE14.4,0FF12.0,1X,1P7E12.4)
        KTAU = MCC(LBU(J),10)
        KSET=IALT-1
        CC 830 K=1,KSET,5
        IF(K.GT.91) GC TO 750
        HT=K+59
        GC TO 760
  750 HT=150+(K-91)*5
  760 C1=HT+6371.
C       THE 6 IN THE CO LCCP MAKES IT CNLY CALCULATE VALUES FCR 0 AND 90 CE
        CC 810 L=1,7,6
        M1=K+1
        PHI=ZANG(L)*.017453292
        SAVTAL=0.C
        STAU(L)=C.0
        SL1A = C.0
        SRAC(L)=C.0
        RANG=0.0
        KSET=IALT
```

47

```
        CC 790 M=M1,MSET
        IF(M.GT.91) GC TO 770
        C2=M+59+6371
        GC TO 780
770     C2=150+(M-91)*5+6371
780     ANG=ASIN(SIN(PHI)*C1/C2)
        CS=SQRT(C1*C1+D2*C2-2.*C1*D2*CCS(PHI-ANG))
        CELS=(CS-RANG)*1.CE+C5
        RANG=CS
        CELTAU = CELS*(0.5*(TAU(M,J)+TAU(M-1,J))
        STAU(L) = STAU(L) + CELTAU
        SL1E = SL1(KTAU,STAU(L))
        RADA = RAC(M-1)/TAU(M-1,J)
        RADE = RAD(M)/TAU(M,J)
        SFAC = (RADP - RACA)/CELTAU
        SRAD(L) = SRAD(L) + SFAC*(SL1A - SL1E)
        SL1A = SL1E
        SAVTAU=STAU(L)
790     CCNTINUE
        SRAD(L) = SRAD(L) + RADE*SLC(KTAU,SAVTAU)
810     CCNTINUE
        SAVTAU=0.C
        STAU(7)=C.C
        SL1A = 0.C
        VRANG = C.C
        SRAD(8)=C.0
        MSET=IALT
        CC 815 M=K,MSET
        L=MSET-M+K
        IF(L.GT.91) GC TO 811
        C2=L+59+6371
        GC TO 812
811     C2=150+(L-91)*5+6371
812     CS=SQRT(C2*C2-D1*C1)

        IF(L.LT.MSET) GO TC 813
        RANG=CS
        GC TO 815
813     CELS=(RANG-CS)*1.CE+C5
        VRANG = VRANG + CELS
        RANG=CS
        SAVTAU = STAU(7)
        CELTAU = CELS*(0.5*(TAU(L,J)+TAU(L+1,J)))
        STAU(7) = STAU(7) + DELTAU
        SL1B = SL1(KTAU,STAU(7))
        RADB = RAD(L)/TAU(L,J)
        RADA = RAC(L+1)/TAU(L+1,J)
        SFAC = (RADB - RACA)/CELTAU
        SRAC(8) = SRAD(8) + SFAC*(SL1C - SL1B)
        SL1A = SL1E
815     CCNTINUE
        M1=K+1
        RANG=0.0
        MSET=IALT
        CC 818 M=M1,MSET
        IF(M.GT.91) GC TO 816
        C2=M+59+6371
        GC TO 817
816     C2=150+(M-91)*5+6371
817     CS=SQRT(C2*C2-D1*C1)
        CELS=(CS-RANG)*1.CE+C5
        VRANG = VRANG + CELS
        RANG=CS
        SAVTAU = STAU(7)
        CELTAU = CELS*(0.5*(TAU(M,J)+TAU(M-1,J)))
        STAU(7) = STAU(7) + DELTAU
        SL1B = SL1(KTAU,STAU(7))
        RADA = RAD(M-1)/TAU(M-1,J)
        RADB = RAD(M)/TAU(M,J)
        SFAC = (RADB -RACA)/CELTAU
        SRAD(8) = SRAD(8) + SFAC*(SL1A - SL1B)
        SL1A = SL1E
818     CCNTINUE
        SRAC(8) = SRAD(8) + RADE*SLC(KTAU,STAU(7))
        WRITE(3,4444) HT,SRAC(8),SRAC(1),SRAC(7)
4444    FORMAT (FE.1,1P3E17.4)
        WRITE(6,7401 HT,SRAC(8), HT,(SRAC(M), M= 1,7)
830     CCNTINUE
840     CCNTINUE
        STCP 77
        END
```

48

```fortran
      FUNCTION SLO (K,TAU)
      T = AES (TAU)
      IF (K .EQ. 2) GC TC 200
      IF (T .GT. 4.0) GC TC 110
      X = (T - 2.0)*0.5
      SLO = (((((C.0004110*X - C.0018496)*X + 0.0068370)*X
     1 - 0.0246060)*X + 0.0792553)*X - 0.2701023)*X + 0.6569377)*T
      RETURN
110   CONTINUE
      IF (T .GT. 10.0) GC TC 120
      X = (T - 7.0)/3.0
      SLO = ((((((0.0000191*X - C.0007254)*X + 0.0023078)*X
     1 - 0.0080348)*X + C.0274865)*X - 0.0942082)*X + 0.3539295)*T
      RETURN
120   CONTINUE
      Y = ALOG(T)
      IF (T .GT. 100.0) GC TC 130
      X = 9.210340371976/Y - 3.0
      SLO = ((((((C.0010485*X - 0.0001530)*X - C.0001490)*X
     1 + 0.0013943)*X - 0.0063060)*X + 0.0837063)*X + 1.1415564)*Y
      RETURN
130   CONTINUE
      X = 9.210340371976/Y - 1.0
      SLO = ((((((((0.007142 - C.006222*X) + 0.005211)*X
     1 - 0.004896)*X - C.0064770)*X + C.007743)*X - 0.015900)*X
     2 + 0.122528)*X + C.940111)*Y
      RETURN
200   CONTINUE
      IF (T .GT. 4.0) GC TC 210
      X = (T - 2.0)*0.5
      SLO = ((((((0.0002431*X - 0.0010901)*X + C.0041830)*X
     1 - 0.0160764)*X + C.0681769)*X - 0.2013270)*X + 0.7188944)*T
      RETURN
210   CONTINUE
      IF (T .GT. 10.0) GC TC 220
      X = (T - 7.0)/3.0
      SLO = ((((((0.0001199*X - C.0004699)*X + C.0016637)*X
     1 - 0.0069345)*X + 0.0255562)*X - 0.1003592)*X + 0.4282944)*T
      RETURN
220   CONTINUE
      Y = ALOG(T)
      X = 4.6051701859880/Y - 1.0
      SLO = (((((((((C.036431 - 0.035826*X)*X + 0.062211)*X
     1 - 0.058855)*X - C.046818)*X + C.045072)*X + 0.009249)*X
     2 - 0.000887)*X - C.062059)*X + C.163319)*X + 1.3997669)*Y
      RETURN
      END

      FUNCTION SL1 (K,TAU)
C
C     IF (K .EQ. 1) COMPUTE FUNCTION VALUE FOR PARALLEL BAND.
C     IF (K .EQ. 2) COMPUTE FUNCTION VALUE FOR PERPENDICULAR BAND.
C
      T = ABS(TAU)
      IF (K .GE. 2) GC TC 200
      IF (T .GT.1.0) GC TC 110
      X = (T - 0.5)*2.0
      SL1 = (((1.19E-5*X - 1.94E26-4)*X + 0.00276149)*X
     1 - C.03563735)*X + C.46139473)*T*T
      RETURN
110   CONTINUE
      IF (T .GT. 4.0) GC TC 120
      X = (T - 2.5)/1.5
      SL1 = ((((3.497E-4 - 5.40E-5*X)*X - 2.0328E-3)*X
     1 + C.0114087)*X - 0.0613564)*X + 0.2531337)*T*T
      RETURN
120   CONTINUE
      IF (T .GT. 10.0) GC TC 130
      X = (T - 7.0)/3.0
      SL1 = ((((7.771E-4 - 1.83E-4*X)*X - 2.9392E-3)*X
     1 + C.0120034)*X - 0.0501136)*X + 0.2354310)*T*T
      RETURN
130   CONTINUE
      IF (T .GT. 100.0) GC TC 140
      Y = ALOG(T)
      X = 9.210340371976/Y - 3.0
      SL1 = (((((6.67E-5*X - 6.76E-4)*X + 0.0021934)*X
     1 + C.0011841)*X - C.0058382)*X + 0.8538769)*T*Y
      RETURN
140   CONTINUE
      IF (T .GT. 1000.0) GC TC 150
      Y = ALOG(T)
```

49

```fortran
      X = 27.631021115S3/Y - 5.C
      SL1 = (((1.636E-4*X - 1.56C4E-3)*X + 1.6852E-3)*X
     1 + 0.8616296)*T*Y
      RETURN
150   CONTINUE
      Y = ALOG(T)
      X = 13.815510557SE/Y - 1.0
      SL1 = ((((((2.56F-4*X + 2.309E-3)*X - 7.600E-3)*X
     1 + 6.917E-3)*X + 2.37E-4)*X - 1.2799E-2)*X
     2 + 0.030224)*X + C.83R671)*T*Y
      RETURN
200   CONTINUE
      IF (T .GT. 1.0)  GC TC 210
      X = (T - 0.5)*2.C
      SL1 = ((((7.09E-6*X - 1.2125E-4)*X + 1.87580E-3)*X
     1 - 0.02832252)*X + C.46S67333)*T*T
      RETURN
210   CONTINUE
      IF (T .GT. 4.C)  GC TC 220
      X = (T - 2.5)/1.5
      SL1 = (((((2.14E-4 - 3.26F-5*X)*X - 1.3290E-3)*X
     1 + 8.3419E-3)*X - 0.0530254)*X + 0.28016941)*T*T
      RETURN

220   CONTINUE
      IF (T .GT. 10.0)  GC TC 230
      X = (T - 7.0)/3.C
      SL1 = (((((5.230E-4 - 1.152E-4*X)*X - 2.2120E-3)*X
     1 + 0.0102527)*X - 0.0494252)*X + C.2718101)*T*T
      RETURN
230   CONTINUE
      IF (T .GT. 100.C)  GC TC 240
      Y = ALOG(T)
      X = 9.21074037197E/Y - 3.0
      SL1 = (((((1.068E-4*X - 1.3224E-3)*X + 3.8304E-3)*X
     1 + 8.184CE-3)*X - 6.6082CF-2)*X + 1.057781)*T*Y
      RETURN
240   CONTINUE
      IF (T .GT. 1000.C)  GC TC 25C
      Y = ALOG(T)
      X = 27.631021115S3/Y - 5.0
      SL1 = (((2.668E-4*X - 2.2012E-3)*X
     1 - 0.01815CE)*X + 1.1468715)*T*Y
      RETURN
250   CONTINUE
      Y = ALOG(T)
      X = 13.815510557SE/Y - 1.0
      SL1 = (((((4.969F-4*X - 2.2191F-3)*X + 3.6173E-3)*X
     1 - 2.8417E-3)*X + 3.3462E-3)*X - 0.0192525)*X
     2 + 3.6151E-3)*X + 1.1747922)*T*Y
      RETURN
      END
```

```
      PROGRAM SPCTRA1 (OUTPUT,TAPE1,TAPE1,TAPE2,TAPE5)
      DIMENSION LSC(25),LCC(25),FCM(25),BV(25),LPC(25),LBL(25),STR(25)
      DIMENSION CCN(201,10),TEMP(201),PROC(201)
      DIMENSION AVTEMP(40,25),EVTEMP(40,25)
      DIMENSION CCLCNT(40,25),HCRCNT(40,25)
      DIMENSION RADA( 50),RADB( 50),RADC( 50)
      COMMON /ISPCCM/ ISP
      READ (1,12) TEMP
      DO 500 NCASES = 1, 2
      REWIND 5
      READ (1,6) NSPTOT
      DO 400 NSP = 1, NSPTOT
      READ (1,6) ISP
      READ (1,41) NLEVEL,NBAND,RM
      DO 210 I = 1, NLEVEL
      READ (1,41) LSC(I),LCC(I),FCM(I),BV(I)
 210  CONTINUE
      DO 220 I = 1, NBAND
      READ (1,41) LPC(I),LBL(I),STR(I)
 220  CONTINUE
      READ (1,6) NALT,NLEV
      DO 221 NZ = 1, NALT
      READ (1,42) (CCN(NZ,NL), NL = 1, NLEVEL)
 221  CONTINUE
C        COMPUTE MEAN TEMPERATURES AND COLUMN COUNTS
      K = NRLEV
      DO 125 NL = 1, NLEVEL
      DO 112 I = 1, NALT
      PROC(I) = CCN(I,NL)*TEMP(I)
 112  CONTINUE
      TP = 0.0
      TC = 0.0
      NA = NALT
      DO 120 I = 1, K
      TP = TP + SIMP(PROC(NA - 2),1.0E+5,2)
     1  + SIMP(PROC(NA - 3),1.0E+5,3) + SIMP(PROC(NA - 5),1.0E+5,3)
      TC = TC + SIMP(CCN(NA - 2,NL),1.0E+5,2)
     1  + SIMP(CCN(NA - 3,NL),1.0E+5,3) + SIMP(CCN(NA - 5,NL),1.0E+5,3)
      CCLCNT(K - I + 1,NL) = TC
      AVTEMP(K - I + 1,NL) = TP/TC
      NA = NA - 5
      HCRCNT(I,NL) = 0.0
 120  CONTINUE
      RBASE = 6.431E+8
      DO 128 I = 1, K
      TPH = 0.0
      TCH = 0.0
      NA = NALT
      RB2 = RBASE*RBASE
      RTOP = 6.431E+8 + NALT*1.0E+5
      SVH = SQRT(RTOP*RTOP - RB2)
      KK = K - I + 1
      DO 125 J = 1, KK
      RTOP = RTOP - 1.0E+5
      STH = SQRT(RTOP*RTOP - RB2)
      H5 = SVH - STH
      SVH = STH

      RTOP = RTOP - 1.0E+5
      STH = SQRT(RTOP*RTOP - RB2)
      H4 = SVH - STH
      SVH = STH
      RTOP = RTOP - 1.0E+5
      STH = SQRT(RTOP*RTOP - RB2)
      H3 = SVH - STH
      SVH = STH
      RTOP = RTOP - 1.0E+5
      STH = SQRT(RTOP*RTOP - RB2)
      H2 = SVH - STH
      SVH = STH
      RTOP = RTOP - 1.0E+5
      STH = SQRT(ABS(RTOP*RTOP - RB2))
      H1 = SVH - STH
      SVH = STH
      TPH = TPH + QINT(PROC(NA - 2),H4,H5,2)
     1  + QINT(PROC(NA - 3),H3,H4,3) + QINT(PROC(NA - 5),H1,H2,3)
      TCH = TCH + QINT(CCN(NA - 2,NL),H4,H5,2)
     1  + QINT(CCN(NA - 3,NL),H3,H4,3) + QINT(CCN(NA - 5,NL),H1,H2,3)
      NA = NA - 5
 125  CONTINUE
      HCRCNT(I,NL) = TCH
      EVTEMP(I,NL) = TPH/TCH
```

51

```fortran
      RBASE = REASE + 5.CE+5
128   CCNTINUE
129   CCNTINUE
      CC 40C NE = 1, NBANC
      CC 230 I = 1, NPLEV
      REAC (1,43) HT,RAEC(I),RACA(I),RADB(I)
230   CCNTINUE
      WRITE(4) (LSC(I),I=1,25)
      WRITE(4) ((HCRCNT(I,J),J=1,25),I=1,40)
      WRITE(4) ((EVTEMP(I,J),J=1,25),I=1,40)
400   CCNTINUE
500   CCNTINUE
      STOP
6     FCRMAT (8IE)
8     FCRMAT (1FEE11.3)
14    FCRMAT (1X)
12    FCRMAT (1FEE12.4)
1E    FCRMAT (1H1)
41    FCFMAT (2IE,5E12.4)
42    FCRMAT (8F10.4)
43    FCRMAT (FE.1,3E17.4)
      END




      FUNCTICN SIMP (A,H,K)
      CIMENSICN A(3)
      IF (K - 2)  100,110,120
100   CCNTINUE
      SIMP = (8.0*A(2) +5.C*A(1) - A(3))*H/12.0
      RETURN
110   CCNTINUE
      SIMP = (8.C*A(2) + 5.C*A(3) - A(1))*H/12.0
      RETURN
120   CCNTINUE
      SIMP = (4.C*A(2) + A(1) + A(3))*H/3.0
      RETURN
      END




      FUNCTICN CINT (A,H1,H2,K)
      CIMENSICN A(3)
      HS = H1 + H2
      IF (K - 2) 100,110,120
100   CCNTINUE
      GINT = ((A(2)*HS - A(1)*H2 - A(3)*H1)*H1/(6.C*H2*HS)
     1 + (A(1) + A(2))*C.5)*H1
      RETURN
110   CCNTINUE
      GINT = ((A(2)*HS - A(1)*H2 - A(3)*H1)*H2/(6.C*H1*HS)
     1 + (A(2) + A(3))*C.5)*H2
      RETURN
120   CCNTINUE
      GINT = (A(2)*HS - A(1)*H2 - A(3)*H1)*(H1**3 + H2**3)
     1 /(6.0*HS*H1*H2) + (A(1)*H1 + A(2)*HS + A(3)*H2)*C.5
      RETURN
      END
```

```
      PROGRAM SPECTRA2 (OUTPUT,TAPE4,TAPE1,TAPE2,TAPE5)
      DIMENSION LEC(25),LSC(25)
      DIMENSION CCN(10),TEMP(201)
      DIMENSION HORCNT(40,25),EVTEMP(40,25)
      DIMENSION RACA( 50),RACB( 50),RADC( 50)
      COMMON /ISPCOM/ ISP
      COMMON WVNLM(2500),ENLS(2500),STPL(2500),HRAC(2500)
      COMMON SPEC(1801)
      DO 600 J=1,14
      REWIND 1
      REWIND 4
      READ (1,12) TEMP
      DO 500 NCASES = 1, 2
      REWIND 5
      DO 106 I = 1, 1801
      SPEC(I) = 0.0
 106  CONTINUE
      READ (1,6) NSPTOT
      DO 400 NSP = 1, NSPTOT
      READ (1,6) ISP
      READ (1,41) NLEVEL,NBAND,RM
      READ(4)(LSC(I),I=1,25)
      READ(4)((HORCNT(I,JKK),JKK=1,25),I=1,40)
      READ(4)((EVTEMP(I,JKK),JKK=1,25),I=1,40)
      DO 210 I = 1, NLEVEL
      READ (1,41) LSCC,LCC,RCM,EV
 210  CONTINUE
      DO 220 I = 1, NBAND
      READ (1,41) LEC(I),LEU,STR
 220  CONTINUE
      READ (1,6) NALT,NRLEV
      DO 221 NZ = 1, NALT
      READ(1,42) (CCN(NL),NL=1,NLEVEL)
 221  CONTINUE
      IF (ISP .NE. 3)  GO TO 225
      CNTHR = 2.0*HORCNT(J,1)
      CALL ROTATE (5,LINES)
C     ROTATE WILL READ FROM TAPE5
      CALL WATSTR (RADC(J),RM,EVTEMP(J,1),CNTHR,1,LINES)
      CALL FILTER (SPEC,HRAC,WVNLM,LINES)
 225  CONTINUE
      DO 400 NE=1,NBAND
      DO 230 I=1,NRLEV
      READ(1,43) HT,RADC(I),RACA(I),RACB(I)
 230  CONTINUE
      NEC=LEC(NE)
      NI=MOD(NEC/100,100)
      NF=MOD(NEC,100)
      IF ((NB .GT. 10) .AND. (ISP .EQ. 3)) GO TO 400

      CALL ROTATE (5,LINES)
      CALL WATSTR (RADC(J),RM,EVTEMP(J,NF),2.0*HORCNT(J,NF),2,LINES)
      CALL FILTER (SPEC,HRAC,WVNLM,LINES)
 400  CONTINUE
      WRITE (2,16)
      WRITE (2,8) SPEC
 500  CONTINUE

 600  CONTINUE
      STOP
   6  FORMAT (6I6)
   8  FORMAT (1P6E11.3)
  14  FORMAT (1X)
  12  FORMAT (1P6E12.4)
  16  FORMAT (1H1)
  41  FORMAT (2I6,5E12.4)
  42  FORMAT (8E10.4)
  43  FORMAT (F6.1,3E17.4)
      END

      SUBROUTINE WATSTR (RAC,RM,T,DEN,N,LINES)
      COMMON /ISPCOM/ ISP
      COMMON WVNLM(2500),ENLS(2500),STR(2500),HRAC(2500)
      TGFAC = (296.0/T)**1.5
      IF (ISP .EQ. 2)  TGFAC = 296.0/T
      IF (ISP .EQ. 4)  TGFAC = 296.0/T
      GO TO (100,200,300), N
 100  CONTINUE
      RADFAC = 1.5806E-20*DEN*TGFAC
      TAUFAC = 4.2815E-13*SQRT(RM/T)*TGFAC*DEN
      DO 110 I = 1, LINES
```

53

```fortran
      HHOLAM = 1.0E+4/WVNLM(I)
      TAU = TALFAC*STR(I)*EXF(-1.43879*ENLS(I)/T)
     1 *(1.0 - EXF(-14387.9/(HHOLAM*T)))
      ROTEA = 2.0*STR(I)/HHCLAM**2
      HRAD(I) = RADFAC*FCTEA*SFUN(TAU)
     1 *EXF(-1.43879*(1.0E+4/HHCLAM + ENLS(I))/T)
 110  CCNTINUE
      RETURN
 200  CCNTINUE
      TAUFAC = 4.8815E-18*SGRT(RM/T)*CEN
      TEFAC = (T - 296.0)/(T*296.0)*1.43879
      SUM = 0.0
      CC 210 I = 1, LINES
      HSTR = TGFAC*STR(I)*EXP(TEFAC*ENLS(I))
      HHOLAM = 1.0E+4/WVNLM(I)
      TAU = TAUFAC*HHOLAM*HSTR
      HRAD(I) = TAL*SFLN(TAL)*EXF(-14387.9/(HHOLAM*T))
     1 /HHCLAM**4
      SUM = SLM + HRAD(I)
 210  CONTINUE
      RADFAC = FAC/SUM
      CC 220 I = 1, LINES
      HRAD(I) = HRAD(I)*RADFAC
 220  CCNTINUE
      RETURN
 300  CCNTINUE
      RETURN
      END


      FLNCTION SFUN (TAL)
      T = ABS(TAL)
      IF (T .GT. 4.0)  GC TC 110
      X = (T - 2.0)*0.5
      SFUN = ((((((0.00100E1*X - 0.004288)*X + 0.013686)*X - 0.042491)
     1 *X + 0.114932)*X - 0.270057)*X + 0.556485
      RETURN
 110  CONTINUE
      IF (T .GT. 10.0)  GC TC 120
      X = (T - 7.0)/3.0
      SFUN = (((((0.000347*X - 0.001187)*X + 0.003148)*X - 0.009545)
     1 *X + 0.028248)*X - 0.081972)*X + 0.247900
      RETURN
 120  CONTINUE
      Y = ALCG(T)
      X = 1.0/Y
      SFUN = ((((((0.06787E*X - 0.286869)*X + 0.264568)*X
     1 - 0.269167)*X + 0.325368)*X + 1.128379)*SQRT(Y)/T
      RETURN
      END


      SUBRCUTTNE FILTEF (SPEC,RAC,WVNUM,N)
C     FIVE WAVENUMBER RESOLUTION, TWC WAVENUMBER INTERVAL
      CIMENSICN SPEC(1801),RAC(450),WVNUM(450)
      CCMMCN /ISFCCM/ ISF
      CC 150 I = 1, N
      RT = WVNLM(I)
      RADT = RAC(I)
      JS = (RT - 405.0)*0.5 + 2.0
      JF = (RT - 395.0)*0.5 + 1.0
      IF (JS .LT. 1)  JS = 1
      IF (JF .GT. 1801)  JF = 1801
      IF (JS .GT. JF)  GC TC 150
      CC 120 J = JS,JF
      RNUM = (J - 1)*2.0 + 400.0
      SPEC(J) = SPEC(J) + (5.0 - AES(RNUM - RT))*RADT*0.04
 120  CCNTINUE
 150  CCNTINUE
      RETURN
      END


      SUBRCUTINE RCTATE (NIN,LINES)
      CCMMCN /ISFCCM/ ISF
      CCMMCN WVNLM(2500),ENLS(2500),STR(2500),HFAC(2500)
      READ (NIN) LINES
      IF (ECF(NIN)) 121,100
 100  READ (NIN)(WVNUM(I),STR(I),ENLS(I),I=1,LINES)
      IF (ECF(NIN))121,115
 119  CCNTINUE
 121  RETURN
      END
```

```
        PROGRAM PLCTK(INPUT,CUTPUT=/5C00,TAPE1=INPUT,TAPE2=CUTPUT,TAPE3)
C       LAST CHANGE FEB 15, 1979


        DIMENSICN K(16)
        DATA COL/*$*/
1       REWINC 3
        READ(3,5CC) TYPE
500     FORMAT(A1,A7,2I8,3F18.11)
        IF(TYPE.EC.COL) GC TO 10
        REWINC 3


        WRITE(2,1000)
1000    FORMAT(* ENTER DELT,START,STCF,AMP,OFFSET,STARTO*)
        READ(1,*) DELT,WINC,XENC,AMF,CFF,START
        GC TO 15
10      REWINC 3
        READ(3,500) TYPE,TITLE,NUM,IREV,AE,DELT,START
        WRITE(2,550)
550     FORMAT(* ENTER START,STCF,AMF,CFFSET*)
        READ(1,*) WIND,XENC,AMF,CFF

15          REWINC 2
        XDELT=RS0./(XENC-WINC)
            XDELT=DELT*XDELT
        WRITE(2,560) TITLE
560     FORMAT(/,15X,A7)
            WRITE(2,20000)WINC,XEND
20000      FORMAT(3X,*INTERVAL BETWEEN *,F8.3,* AND *,F8.3)
            WRITE(2,30000) CFF,AMF
30000   FORMAT(3X,*AMPLITUDE RANGE(*,F7.0,* TO *,F7.0,*)*)

        REWINC 2
        CALL FLOTS
        CALL FRAME




        IY2=8
        IX2=10
        X2=10
        XN=START

        CO 100 I=1,100000
        REAC(3,40000)(K(J),J=1,16)
40000      FORMAT(16I5)
        IF(ECF(3)) 900,80,80
80      X=XN
        XN=XN+16*DELT
        IF(XN.LT.WINC)GCTC 100
        CO 90 J=1,16
        XJ=X+FLOAT(J-1)*CELT
        IF(XJ.LT.WINC)GCTC 90



        IX=IX2
        IY=IY2
        X2=X2+XDELT
        IX2=X2
            IF(IX2.GT.900)GCTC 800
        IY2=500.*(K(J)-CFF)/(AMP-CFF)+8.5
        IF(IY2.LE.1) IY2=1
        CALL LINE(IX,IY,IY2,IY2)
90      CONTINUE
100     CONTINUE

800     REWIND 3
        CALL XMTT
        READ(1,*)JUNK
        REWINC 2
        GOTC 1
900     STOP 7
        END
```

55

```
C     THIS ROUTINE INITIALIZES THE PLOT BUFFER
C     AND THE INDEXES INTO THE PLOT BUFFER.
C
      SUBROUTINE PLOTS
      COMMON/IXY/IT,ICX,ICLY,ICHY,IOLX,IOHX
      COMMON/BUFFER/LB,LW,IB(500)

      LB=1
      IT=0
      LW=36
      IB(1)=0007000000000000000B
      RETURN
      END
```

```
C
C     THIS SUBROUTINE PACKS A 12-BIT
C     PLOT INSTRUCTION FOR THE TEKTRONIX INTO
C   THE BUFFER (IB). IT IS FORMATTED TO THE
C   PECULIAR BIT STREAM REQUIREMENTS FOR TEMPO.
      SUBROUTINE SEND(I)
      COMMON/BUFFER/LB,LW,IB(500)
      IF(LW .GE. 0)GOTO 2
      LW=48
      LB=LB+1
2     IB(LB)=OR(AND(IB(LB),SHIFT(MASK(48),LW)),SHIFT(I,LW))
      LW=LW-12
1     FORMAT(3X,I4,3X,I4)
      RETURN
      END
```

```
C
C
C
      SUBROUTINE XMIT
      COMMON/IXY/IT,ICX,ICLY,ICHY,IOLX,IOHX
      COMMON/BUFFER/ LB,LW,IB(500)
      CALL SEND(IOHY)
      CALL SEND(IOLY)
      CALL SEND(IOHX)
      CALL SEND(IOLX)
      DO 1 K=1,9
      IF(LW.EQ.0)GOTO 2
1     CALLSEND(0)
2     BUFFER OUT(2,0) (IB(1),IB(LB))
      IF(UNIT(2)) 3,3,3
3     CALL PLOTS
      RETURN
      END
```

```
C
      SUBROUTINE PAGE

      CALL SEND(4033B)
      CALL SEND(4014B)
      CALL SEND(4007B)
      CALL SEND(0)
      CALL XMIT
      RETURN
      END
```

```
C
C
        SUPPOUTINE COPY

        CALL SEND(4033E)
        CALL SEND(4027E)
        CALL XPIT
        RETURN
        END


C
          SUBROUTINE FRAME
      CALL LINE(10,509,900,509)
        CALL LINE(10,766,900,766)
        CALL LINE(10,8,900,8)
        CALL LINE(9,768,9,9)
        CALL LINE(898,768,898,9)
          CALL LINE(10,768,900,768)
          CALL LINE(10,767,900,767)
          CALL LINE(10,10,900,10)
          CALL LINE(10,9,900,9)
          CALL LINE(10,768,10,9)
          CALL LINE(11,768,11,9)
          CALL LINE(900,768,900,9)
          CALL LINE(899,768,899,9)
      RETURN
      END


C
          SUBROUTINE LINE(J,I,JJ,IT)
        COMMON/BUFFER/LB
      CALL SEND(4035B)

        K=OR(AND(SHIFT(I,-5),-MASK(55)),4040B)
        CALL SEND(K)
        K=OR(AND(I,-MASK(55)),4140E)
        CALL SEND(K)
        K=OR(AND(SHIFT(J,-5),-MASK(55)),4040B)
        CALL SEND(K)
        K=OR(AND(J,-MASK(55)),4100B)
        CALL SEND(K)
        K=OR(AND(SHIFT(II,-5),-MASK(55)),4040B)
        CALL SEND(K)
        K=OR(AND(II,-MASK(55)),4140E)
        CALL SEND(K)
        K=OR(AND(SHIFT(JJ,-5),-MASK(55)),4040B)
        CALL SEND(K)
        K=OR(AND(JJ,-MASK(55)),4100E)
        CALL SEND(K)

        IF(LB.GT.480)CALL XMIT

          RETURN
          END
```

```
                                       :*************
                                       :***
                                       :***
        PLOTR  EQU    0F5H             :
        PLTCS  EQU    0F7H             :
        MDMCS  EQU    0EFH
        MDMDT  EQU    0EEH
        TXRDY  EQU    01
        RXRDY  EQU    2
        BMOD0  EQU    80H
        EOCHR  EQU    83H              :
        NDCHR  EQU    0FFH
        LFTPN  EQU    40H              :
        DRPEN  EQU    80H              :
        LWAIT  EQU    50               : TIME DELAY FOR PEN UP/DOWN
        SWAIT  EQU    255
        OVRIT  EQU    10H
        DICMP  EQU    100100018        :
        NXORG  EQU    0600             :
                                       :**
                                       :**
               ORG    0800H
                                       :**
                                       :**
               IN     MDMDT           : CLEARS MODEM
               MVI    A,  'R'          : CYBER RNH COMMAND
               CALL   XMIT
               MVI    A,  'N'
               CALL   XMIT
               MVI    A,  'H'
               CALL   XMIT
               CALL   SNDCR            : XMITS 'RETURN'
               CALL   HEXIN            : IGNORE CR,LF:WAIT FOR PROMPT
                                       :**
               MVI    A,  BMOD0        : INIT PLOTTER PORT

        SET55  EQU    $
               OUT    PLTCS
                                       :**
               LXI    H,  BUFER        : RESET PNTER
               SHLD   PNTER
                                       :**
               LXI    H,  0            : CLEAR CURRX, CURRY
               SHLD   CURRX
               SHLD   CURRY
                                       :**
        MNITR  EQU    $
               CALL   NXCHR            : GET COMMAND
               ADD    A                : OFFSET = 2*COMMAND
                                       :**
               MOV    C,  A
               MVI    B,  0
                                       :**
               LXI    H,  COMND        : GET ADDRESS OF
               DAD    B                : COMMAND ROUTINE
                                       :**
               MOV    E,  M            : INTO DE, THEN
               INX    H
               MOV    D,  M
               XCHG                    : INTO HL, THEN
               PCHL                    : INTO PC
                                       :**
        COMND  EQU    $
               DW     PNLFT            : MOVE W/ PEN UP


               DW     PNDRP            : MOVE W/ PEN DOWN
               DW     XGRAF            : X-AXIS GRAPH ROUTINE
               DW     YGRAF            : Y-AXIS GRAPH ROUTINE
               DW     TXPLT            : PLOT CHARACTERS
               DW     RETCR            : RETURN CURR. X,Y
               DW     SETCR            : RESET CURR. X,Y
               DW     ENDIT            : TERMINATES PLOTTING
                                       :**
               :      COMMAND ROUTINES
```

58

```
                                    ;**
                                    ;**
PNLFT   EQU   $                     ;  #
        CALL  LIFT                  ;  # LIFTS PEN IF NEC
        JMP   PNMOV                 ;  #
;**
PNDRP   EQU   $                     ;  #
        CALL  DROP                  ;  DROPS PEN IF NEC
;
PNMOV   EQU   $                     ;  #
        CALL  REGIN                 ;  #
        SHLD  ENDX                  ;  #
        CALL  REGIN                 ;  #
        SHLD  ENDY                  ;  READS ENDPTS AND

        CALL  PLOTS                 ;  MOVES PEN THERE
                                    ;**
        JMP   MNITR                 ;  #
;**
;**
XGRAF   EQU   $                     ;  PLOT X-AXIS GRAPH
        CALL  NXCHR                 ;  GET X-INCR.
        MOV   C, A                  ;  B IS CLEAR
        LHLD  CURRX                 ;  MOVE PEN TO 1ST POINT
        SHLD  ENDX
        CALL  REGIN
        SHLD  ENDY
        CALL  LIFT
        CALL  PLOTS
LOOPX   EQU   $                     ;  PLOT STREAM
        CALL  REGIN                 ;  GET NEXT Y POINT
        MOV   A, H                  ;  CHECK FOR END OF STREAM
        CPI   NOCHR
        JZ    MNITR
                                    ;**
        SHLD  ENDY
                                    ;**
        LHLD  CURRX                 ;  UPDATE X COORD
        DAD   B
        SHLD  ENDX
                                    ;**
        CALL  DROP                  ;  INSURE PEN DOWN
        CALL  PLOTS
                                    ;**
        JMP   LOOPX
                                    ;**
                                    ;**
                                    ;**
YGRAF   EQU   $                     ;  PLOT Y-AXIS GRAPH
        CALL  NXCHR                 ;  GET Y-INCR.
        MOV   C, A                  ;  B IS CLEAR
        LHLD  CURRX                 ;  MOVE PEN TO 1ST POINT
        SHLD  ENDX
        CALL  REGIN
        SHLD  ENDY


        CALL  LIFT
        CALL  PLOTS
LOOPY   EQU   $                     ;  PLOT STREAM
        CALL  REGIN                 ;  GET NEXT Y POINT
        MOV   A, H                  ;  CHECK FOR END OF STREAM
        CPI   NOCHR
        JZ    MNITR
                                    ;**
        SHLD  ENDY
                                    ;**
        LHLD  CURRX                 ;  UPDATE X COORD
        DAD   B
        SHLD  ENDX
                                    ;**
        CALL  DROP                  ;  INSURE PEN DOWN
        CALL  PLOTS
                                    ;**
        JMP   LOOPY
                                    ;**
                                    ;**
TXPLT   EQU   $
        CALL  NXCHR                 ;  GET SCALE
```

```
              MOV   E,  A         ; AND SAVE IN E
                                  ;**
              CALL  NXCHR         ; GET DIRECTION
              STA   DIRN
                                  ;**
                                  ;**
       CLOOP  EQU   $
              CALL  NXCHR         ; GET NEXT CHAR. CODE
              CPI   NOCHR         ; RETURN TO MONITOR
              JZ    MNITR         ; AT END OF STRING
                                  ;**
              LHLD  CURRX         ; RESET ORIGINS
              SHLD  ORIGX         ; #
              LHLD  CURRY         ; #
              SHLD  ORIGY         ; #
              LXI   H,  CHTAB     ; DATA TABLE FOR CHAR GEN
              ADD   A             ; OFFSET=CODE*2
              JNC   ADOFF
                                  ;**
              INX   H
       ADOFF  EQU   $
              ADD   L             ; A CONTAINS OFFSET
              MOV   L,  A         ; #
              JNC   GTSEG         ; #HL POINTS INTO CHTAB
              INR   H             ; #
       GTSEG  EQU   $
              MOV   C,  M         ; BC POINTS TO SEGS
              INX   H
              MOV   B,  M
       ;**
       SLOOP  EQU   $             ; #
              LDAX  B             ; GET NEXT SEGMENT CODE
              MOV   D,  A         ; SAVE SEG. CODE IN  D
              CALL  CHGEN         ; PLOT SEGMENT
              INX   B             ; INDEX OF NEXT SEGMENT
              MOV   A,  D         ; #
              ANI   EOCHR         ; #CHECK IF LAST SEGMENT
              JZ    SLOOP         ; CONT IF NOT LAST SEG
              MVI   A,  NXORG     ; #PEN UP,  (0,6) * SCALE
              MOV   D,  A
              CALL  CHGEN         ; INTERLETTER SPACE
              JMP   CLOOP         ; FOR NEXT CHAR
                                  ;**


                                  ;**
       RETCR  EQU   $
              LHLD  CURRX         ; SEND CURRX,CURRY
              CALL  RGOUT
              LHLD  CURRY
              CALL  RGOUT
                                  ;**
              CALL  SNDCR         ; SEND 'RET'
              CALL  HEXIN         ; WAIT FOR PROMPT
              JMP   MNITR
                                  ;**
                                  ;**
       SETCR  EQU   $
              CALL  REGIN
              SHLD  CURRX
              CALL  REGIN
              SHLD  CURPY
              JMP   MNITR
                                  ;**
                                  ;**
       ENDIT  EQU   $
              RST   1             ; RETURN TO MONITOR
                                  ;**
                                  ;**
       ;      I/O UTILITIES
                                  ;**
       REGIN  EQU   $             ; 4 HEX DIGITS INTO HL
              CALL  NXCHR
              MOV   H,  A
              CALL  NXCHR
              MOV   L,  A
              RET
                                  ;**
```

60

```
                                      :**
NXCHR EQU   $                 ; GETS HEX CONSTANT FROM BUFFER
      PUSH  B
      PUSH  D
      PUSH  H
      LHLD  PNTER             ; CHECK IF BUFFER EMPTY
      LXI   D,  BUFER
      CALL  CMPAR
      CZ    LOADR             ; FILL IT IF NECESSARY
                                      :**
      MOV   B,  M             ; GET CHARACTER
      INX   H
                                      :**
      PUSH  H
      LXI   D,  BUFER
      LHLD  BUFLN
      DAD   D
      XCHG
      POP   H
                                      :**
      CALL  CMPAR
      JNZ   LDONE
                                      :**
      LXI   H,  BUFER         ; ELSE RESET PNTER
LDONE EQU   $
      SHLD  PNTER
      MOV   A,  B
      POP   H
      POP   D
      POP   B
      RET



                                      :**
                                      :**
LOADR EQU   $
      CALL  LIFT              ; PREVENTS INK SPOTS
      PUSH  D
      PUSH  H
      LXI   H,  BUFER
      CALL  SHAKE
                                      :**
      CALL  HEXIN             ; GET WORD COUNT
      MOV   D,  A
      CALL  HEXIN
      MOV   E,  A
      XCHG  :                 SAVE BUFFER LENGTH
      SHLD  BUFLN
      XCHG
                                      :**
LODNG EQU   $
      CALL  HEXIN             ; GET NEXT WORD
      MOV   M,  A
      INX   H
      DCX   D
      XRA   A                 ; CHECK IF LAST WORD
      ORA   E
      JNZ   LODNG
      ORA   D
      JNZ   LODNG             ; REPEAT UNTIL DONE
      CALL  HEXIN             ; WAIT FOR PROMPT
                                      :**
      POP   H
      POP   D
      RET

                                      :**
                                      :**
                                      :**
                                      :**
                                      :**
HEXIN EQU   $                 ; 2 HEX DIGITS INTO H
      PUSH  B
      CALL  ASCHX             ; ASCII TO HEX CONVERSION
      RLC
      RLC
      RLC
      RLC
      MOV   B,  A             ; SAVE HIGH CHARACTER
      CALL  ASCHX             ; GET LOW CHARACTER
```

61

```
        ORA  B              : THEN ADD
        POP  B
        RET
                            :++
                            :++
                            :++
ASCHX EQU  $                : ASCII TO HEX IN A
        CALL RECEV          : GET ASCII CHAR
        CPI  'A'
        JC   ARNDA          : JUMP IF NUMBER
                            :++
        ADI  9              : ADJUST A-F
ARNDA EQU  $
        ANI  0FH            : CLEAR HIGH BITS
        RET
                            :++
                            :++
                            :++


RECEV EQU  $
        IN   MDMCS          : TEST MODEM STATS
        ANI  RXRDY
        JZ   RECEV          : WAIT FOR CHAR
                            :++
        IN   MDMDT          : READ CHAR
        ORA  A
        JZ   RECEV          : IGNORE NULL CHARS
        CPI  0AH            :       ...LF &
        JZ   RECEV
        CPI  0DH            :       ...CR
        JZ   RECEV
        RET
                            :++
                            :++
                            :++
RGOUT EQU  $                : (HL) TO 4 ASCII CHARS
        CALL HXOUT
        MOV  H,  L
        CALL HXOUT
        RET
                            :++
                            :++
                            :++
HXOUT EQU  $                : (H) TO 2 ASCII CHARS
        MOV  A,  H
        RRC
        RRC
        RRC
        RRC
        CALL HXASC
        MOV  A,  H          : LOW 4 BITS
        CALL HXASC
        RET
                            :++
                            :++
                            :++
HXASC EQU  $                : (A) TO 1 ASCII CHAR
        ANI  0FH            : SEND LOW 4 BITS
        CPI  0AH
        JC   ARNDC          : CHECK FOR AND
                            :++
        ADI  7              : ADJUST A - F
ARNDC EQU  $
        ADI  '0'            : FOR ASCII "PREFIX"
        CALL XMIT
        RET
                            :++
                            :++
                            :++
XMIT  EQU  $
        PUSH B
        MOV  C,  A
                            :++
XLOOP EQU  $
        IN   MDMCS
        ANI  TXRDY
        JZ   XLOOP
```

62

```
                          :•••
        MOV   A,  C
        OUT   MDMDT
        CALL  RECEV
        POP   B
        RET
                          :••

                          :••
                          :••
SHAKE   EQU   $
        PUSH  H               : WAIT FOR CYBER FOX
        MVI   L,   CAH        : APPROX. 1 SEC
EWAIT   EQU   $
        CALL  LONG
        DCR   L
        JNZ   EWAIT

        POP   H

        MVI   A,   '1'        : HANDSHAKE CHAR
        CALL  XMIT
                          :••
SNDCR   EQU   $
        IN    MDMCS           : XMITS 'CF'
        ANI   TXRDY
        JZ    SNDCR
                          :••
        MVI   A,   0DH        : 'RETURN'
        OUT   MDMDT           : ECHO & LF IGNORED
        RET
                          :••
                          :••
        :     PEN CONTROL UTILITIES
                          :••
LIFT    EQU   $          :  #
        LDA   PSTAT      :  #
        ORA   A          :  #
        RZ               :  #
        MVI   A,  LFTPN  :
        CALL  MOVE       :  CONTROLS PLOTTER
        JMP   PWAIT      :  #

DROP    EQU   $          :  #
        LDA   PSTAT      :  #
        ORA   A          :  #
        RNZ              :  #
        MVI   A,   DRPEN :  #••••••••••
        CALL  MOVE       :  CONTROLS PLOTTER
        INR   A          :  PSTAT GETS 1

PWAIT   EQU   $          :  #
        STA   PSTAT      :  #
LONG    EQU   $          :  #
        PUSH  H          :  #
        MVI   L,   LWAIT :  #
                          :••
LLOOP   EQU   $          :  #
        CALL  SHORT      :  #
        DCR   L          :  #
        JNZ   LLOOP      :  #
                          :••
        POP   H          :  #
        RET              :  #
                          :••
                          :••
                          :••
MOVE    EQU   $
        OUT   PLOTR          : SEND INSRTUCTION
        XRA   A
        OUT   PLOTR          : FOLLOWED BY 0
        CALL  SHORT          : WAIT 1.65 MILLISEC
```

```
                                        :**
        SHORT EQU   $               : #
              PUSH  H
              MVI   L,  SWAIT
                                        :**
        WLOOP EQU   $
              DCR   L
              JNZ   WLOOP
              POP   H
              RET                   : #
                                        :**
              :    ARITHMETIC UTILITIES
                                        :**
        DPSUB EQU   $               : #
              MOV   A,  E           : DOUBLE PRECISION
              SUB   L               : SUBTRACTION ROUTINE
              MOV   L,  A           : #
              MOV   A,  D           : #
              SBB   H               : #
              MOV   H,  A           : #
              XRA   A               : RESULT IN HL,  ACCUM CLEARED
              RET                   : #
                                        :**
        CMPAR EQU   $               : #
              MOV   A,  D           : DOUBLE PRECISION COMPARISON ROUTIN(
              CMP   H               : #
              RNZ                   : #HIGH BYTES NE 0
              MOV   A,  E           : #
              CMP   L               : #
              RET                   : #
                                        :**
        SCALR EQU   $               : #
              ANI   7               : #
                                        :**
        MULT  EQU   $               : 8 X 8 MULTIPLICATION
              PUSH  B               : ROUTINE
              PUSH  D               : HL GETS A*E
              MVI   C,  8
              MVI   D,  0
              MOV   L,  D           : #
              MOV   H,  L           : #
                                        :**
        MLOOP EQU   $               : #
              RAR                   : #
              MOV   B,  A           : #
              JNC   ARNDM           : #
              DAD   D               : #
              CMC                   : #
                                        :**
        ARNDM EQU   $               : #
              XCHG                  : SHIFT MULTIPLICAND LEFT
              DAD   H               : #
              XCHG
              DCR   C               : #
              JNZ   MLOOP           : #
                                        :**
              POP   D
              POP   B
              RET                   : #
                                        :**
              :    MAIN-LINE UTILITIES
                                        :**


                                        :**
        CHGEN EQU   $               : #
              PUSH  B               : SAVE REGS.
              PUSH  D
              PUSH  H
                                        :**
              ANI   1000            : PEN UP/DOWN FIELD
              JNZ   CHKDN           : DECODE PEN UP/DOWN
                                        :**
              CALL  LIFT            : #
              JMP   RELPT           : #
                                        :**
        CHKDN EQU   $               : #
              CALL  DROP            : #
```

64

```
RELPT  EQU   $                    ; #
       MOV   A,   D               ; D CONTS SEG CODE
       CALL  SCALR                ; (A.AND.7)*SCALE   (E)
       MOV   B,   H               ; LOADS  BC WITH
       MOV   C,   L               ; RELATIVE Y
;
       MOV   A,   D               ; #
       RRC                        ; #
       RRC                        ; #
       RRC                        ; SHIFT FOR RELATIVE X
       CALL  SCALR                ;     IGNORES HIGH BITS IN  A
       XCHG                       ; STORE REL X IN  DE
;
       LDA   DIRN                 ; COMPUTE DIRECTION OF TEXT
       MOV   H,   A               ; #
       ORA   A                    ; SETS FLAGS
       JPE   DIRX                 ; JUMP IF ON X-AXIS (0&3)
;
       PUSH  B                    ; SWAP X AND Y VALUES
       MOV   B,   D               ; FOR VERTICAL LETTERS
       MOV   C,   E               ;      (1&2)
       POP   D
;**
DIRX   EQU   $                    ; #
       ANI   10B                  ; #
       JZ    DIRY                 ; #
;
       XRA   A                    ; COMP X VALUE
       SUB   E                    ; #
       MOV   E,   A               ; #
       MVI   A,   0               ; #
       SBB   D                    ; #
       MOV   D,   A               ; DE GETS  -DE
;
DIRY   EQU   $                    ; #
       MOV   A,   H               ; #
       ANI   01B                  ; #
       JZ    SETPT                ; #
;
       XRA   A                    ; COMP Y VALUE
       SUB   C                    ; #
       MOV   C,   A               ; #
       MVI   A,   0               ; #
       SBB   B                    ; #
       MOV   B,   A               ; BC GETS  -BC
;**
SETPT  EQU   $                    ; #
;
       LHLD  ORIGX                ; #
       DAD   D                    ; #
       SHLD  ENDX                 ; #


;
       LHLD  ORIGY                ; #
       DAD   B                    ; #
       SHLD  ENDY                 ; #
;
       CALL  PLOTS                ; #
;
       POP   H                    ; RESTORE REGS.
       POP   C
       POP   B
                                  ;**
       RET                        ; #
                                  ;**
                                  ;**
CHTAB  EQU   $                    ; DATA FOR CHAR GEN
       DW    CHR26                ; CHAR:  !
       DW    CHR33                ; CHAR:  A
       DW    CHR34                ; CHAR:  B
       DW    CHR35                ; CHAR:  C
       DW    CHR36                ; CHAR:  D
       DW    CHR37                ; CHAR:  E
       DW    CHR38                ; CHAR:  F
       DW    CHR39                ; CHAR:  G
       DW    CHR40                ; CHAR:  H
       DW    CHR41                ; CHAR:  I
       DW    CHR42                ; CHAR:  J
```

65

```
         DW      CHR43           :  CHAR:  K
         DW      CHR44           :  CHAR:  L
         DW      CHR45           :  CHAR:  M
         DW      CHR46           :  CHAR:  N
         DW      CHR47           :  CHAR:  O
         DW      CHR48           :  CHAR:  P
         DW      CHR49           :  CHAR:  Q
         DW      CHR50           :  CHAR:  R
         DW      CHR51           :  CHAR:  S
         DW      CHR52           :  CHAR:  T
         DW      CHR53           :  CHAR:  U
         DW      CHR54           :  CHAR:  V
         DW      CHR55           :  CHAR:  W
         DW      CHR56           :  CHAR:  X
         DW      CHR57           :  CHAR:  Y
         DW      CHR58           :  CHAR:  Z
         DW      CHR16           :  CHAR:  0
         DW      CHR17           :  CHAR:  1
         DW      CHR18           :  CHAR:  2
         DW      CHR19           :  CHAR:  3
         DW      CHR20           :  CHAR:  4
         DW      CHR21           :  CHAR:  5
         DW      CHR22           :  CHAR:  6
         DW      CHR23           :  CHAR:  7
         DW      CHR24           :  CHAR:  8
         DW      CHR25           :  CHAR:  9
         DW      CHR11           :  CHAR:  +
         DW      CHR13           :  CHAR:  -
         DW      CHR10           :  CHAR:  *
         DW      CHR15           :  CHAR:  /SLASH
         DW      CHR08           :  CHAR:  (
         DW      CHR09           :  CHAR:  )
         DW      CHR04           :  CHAR:  $
         DW      CHR29           :  CHAR:  =
         DW      CHR69           :  CHAR:   BLANK
         DW      CHR12           :  CHAR:  ,
         DW      CHR14           :  CHAR:  .
         DW      CHR03           :  CHAR:  #POUND SIGN
         DW      CHR59           :  CHAR:  [OPEN SQ BRKT


         DW      CHR61           :  CHAR:  ]
         DW      CHR05           :  CHAR:  PERCENT
         DW      CHR02           :  CHAR:  "DBL. QUOTE
         DW      CHR63           :  CHAR:  _UNDERSCORE
         DW      CHR01           :  CHAR:  EXCLAMATION
         DW      CHR06           :  CHAR:  &AMPERSAND
         DW      CHR07           :  CHAR:  'APOSTROPHIE
         DW      CHR31           :  CHAR:  ?QUESTION MARK
         DW      CHR28           :  CHAR:  <
         DW      CHR30           :  CHAR:  >
         DW      CHR32           :  CHAR:  @COMMERCIAL AT
         DW      CHR60           :  CHAR:  \BACK SLASH
         DW      CHR62           :  CHAR:  ^CIRCUMFLEX
         DW      CHR27           :  CHAR:  :
         DW      CHR64           :  CHAR:  APOSTROPHIE
         DW      CHR65           :  CHAR:  OPEN PAREN
         DW      CHR66           :  CHAR:  VERTICAL LINE
         DW      CHR67           :  CHAR:  CLOSE PAREN
         DW      CHR68           :  CHAR:  TILDE
                                 :**
CHLST   EQU     $               :  CHARACTER TABLES
                                 :**
CHR01   EQU     $               :  CHAR:
        DB      0160
        DB      1120
        DB      0110
        DB      3100
                                 :**
CHR02   EQU     $               :  CHAR:  "
        DB      0140
        DB      1260
        DB      0340
        DB      3460
                                 :**
CHR03   EQU     $               :  CHAR:  #
        DB      0130
```

66

```
                DB      1260
                DB      0360
                DB      1230
                DB      0350
                DB      1150
                DB      0140
                DB      3340
                                        ;**
        :
        CHR04 EQU       $               : CHAR: $
                DB      0020
                DB      1110
                DB      1310
                DB      1420
                DB      1330
                DB      1130
                DB      1040
                DB      1150
                DB      1350
                DB      1440
                DB      0260
                DB      3200
                                        ;**
        :
        CHR05 EQU       $               : CHAR: PERCENT
                DB      1460
                DB      1160


                DB      1050
                DB      1140
                DB      1250
                DB      1160
                DB      0410
                DB      1320
                DB      1210
                DB      1300
                DB      3410
                                        ;**
        :
        CHR06 EQU       $               : CHAR: &
                DB      0420
                DB      1200
                DB      1100
                DB      1010
                DB      1020
                DB      1350
                DB      1260
                DB      1150
                DB      3400
                                        ;**
        :
        CHR07 EQU       $               : CHAR: '
                DB      0260
                DB      3240
                                        ;**
        :
        CHR08 EQU       $               : CHAR: (
                DB      0360
                DB      1240
                DB      1220
                DB      3300
                                        ;**
        :
        CHR09 EQU       $               : CHAR: )
                DB      0160
                DB      1240
                DB      1220
                DB      3100
                                        ;**
        :
        CHR10 EQU       $               : CHAR: *
                DB      0130
                DB      1350
                DB      0250
                DB      1230
                DB      0150
                DB      3330
                                        ;**
```

67

```
CHR11 EQU    $              ; CHAR: +
      DB     0130
      DB     1330
      DB     0240
      DB     3220
                            ;**

CHR12 EQU    $              ; CHAR: ,
      DB     0220
      DB     3100
                            ;**

CHR13 EQU    $              ; CHAR: -
      DB     0130


      DB     3330
                            ;**

CHR14 EQU    $              ; CHAR: .
      DB     0110
      DB     3100
                            ;**

CHR15 EQU    $              ; CHAR:
      DB     3460
                            ;**

CHR16 EQU    $              ; CHAR: 0
      DB     0100
      DB     1020
      DB     1040
      DB     1160
      DB     1360
      DB     1440
      DB     1420
      DB     1300
      DB     3100
                            ;**

CHR17 EQU    $              ; CHAR: 1
      DB     0140
      DB     1260
      DB     3200
                            ;**

CHR18 EQU    $              ; CHAR: 2
      DB     0050
      DB     1160
      DB     1360
      DB     1450
      DB     1440
      DB     1120
      DB     1000
      DB     3400
                            ;**

CHR19 EQU    $              ; CHAR: 3
      DB     0010
      DB     1100
      DB     1330
      DB     1410
      DB     1420
      DB     1330
      DB     1230
      DB     0330
      DB     1440
      DB     1450
      DB     1360
      DB     1160
      DB     3050
                            ;**

CHR20 EQU    $              ; CHAR: 4
      DB     0300
      DB     1360
      DB     1020
      DB     3420
```

68

```
                                ;**
CHR21  EQU    $                 ; CHAR: 5

       DB     0010
       DB     1100
       DB     1300
       DB     1410
       DB     1430
       DB     1340
       DB     1140
       DB     1030
       DB     1060
       DB     3460
                                ;**
CHR22  EQU    $                 ; CHAR: 6
       DB     0030
       DB     1140
       DB     1340
       DB     1430
       DB     1410
       DB     1300
       DB     1100
       DB     1010
       DB     1050
       DB     1160
       D9     1360
       DB     3450
                                ;**
CHR23  EQU    $                 ; CHAR: 7
       DB     0060
       DB     1460
       DB     3100
                                ;**
CHR24  EQU    $                 ; CHAR: 8
       DB     0100
       DB     1010
       DB     1020
       DB     1130
       DB     1040
       DB     1050
       DB     1160
       DB     1360
       DB     1450
       DB     1440
       DB     1330
       DB     1130
       DB     0330
       DB     1420
       DB     1410
       DB     1300
       DB     3100
                                ;**
CHR25  EQU    $                 ; CHAR: 9
       DB     0010
       D9     1100
       D9     1300
       DB     1410
       DB     1450
       D9     1360
       D9     1160
       DB     1050
       DB     1030
       DB     1120
       DB     1320


       DB     3430
                                ;**
CHR26  EQU    $                 ; CHAR: :
       DB     0140
       DB     1130
```

69

```
            DB    0120
            DB    3110
                                    ;**
    :
    CHR27 EQU   $              : CHAR: :
            DB    0240
            DB    1230
            DB    0220
            DB    3100
                                    ;**
    :
    CHR28 EQU   $              : CHAR: <
            DB    0460
            DB    1030
            DB    3400
                                    ;**
    :
    CHR29 EQU   $              : CHAR: =
            DB    0040
            DB    1440
            DB    0020
            DB    3420
                                    ;**
    :
    CHR30 EQU   $              : CHAR: >
            DB    1430
            DB    3060
                                    ;**
    :
    CHR31 EQU   $              : CHAR: ?
            DB    0050
            DB    1160
            DB    1260
            DB    1350
            DB    1340
            DB    1130
            DB    1120
            DB    0110
            DB    3100
                                    ;**
    :
    CHR32 EQU   $              : CHAR: @
            DB    0330
            DB    1220
            DB    1130
            DB    1240
            DB    1340
            DB    1320
            DB    1430
            DB    1440
            DB    1350
            DB    1150
            DB    1040
            DB    1020
            DB    1110
            DB    1310
            DB    3420
                                    ;**
            END1  EQU          $


    :
                                    ;**
                                    ;****
                                    ;****
            ORG   GC00H
                                    ;**
                                    ;**
    PLOTS EQU   $              :  #
            PUSH  3
            PUSH  D
            PUSH  H
    :
            LXI   B,  C        : CONVIENT CONSTANTS
    :
            LHLD  CURFX        : #
            XCHG               : LOADS DE WITH CURFX AND RESETS CURR
            LHLD  ENDX
            SHLD  CURPX        : #
```

70

```
              XCHG
      :
              CALL  CMPAR              : COMPARES (DE) TO (HL)
              JC    NEGDX              : ENDX<CURRX
              JN7   POSDX              : ENDX>CURRX
      :
NODX: CALL  YCOMP                      : COMPARES ENDY AND CURRY,  LOADS ACC
                                       #        AND THEN CLEARS B
              JC    QUAD4              : ENDY<CURRY
              JNZ   QUAD2              : ENDY>CURRY
              JMP   WRAP               : ENDY=CURRY
      :
POSDX EQU   $                         : #
              CALL  YCOMP              : #
              JC    QUAD4              : ENDY<CURRY
              JMP   QUAD1              : ENDY>=CURRY
      :
NEGDX EQU   $                         : #
              XCHG                     : KEEPS DX POSITIVE
              CALL  YCOMP              : #
              JC    QUAD3              : ENDY<CURRY
              JNZ   QUAD2              : ENDY>CURRY
              JMP   QUAD3              : ENDY=CURRY
      :
QUAD4:RLC                             : #
              INR   B                  : #
QUAD3:RLC                             : #
              INR   B                  : #
              XCHG                     : KEEPS DY POSITIVE
QUAD2:RLC                             : #(DE)=GTR(ENDY,  CURRY)
              INR   B                  : #(HL)=LSR(CURRY,  ENDY)
QUAD1:RLC                             : #
              INR   B                  : #
              ANI   0FH                : USE LOWEST 4 BITS
              STA   DIAG               : COMPUTES DY=(DE)-(HL)
              CALL  DPSUB              : COMPUTES DY  (DE)-(HL)
              SHLD  DY                 : #       AND CLEARS ACCUM.
      :
              MOV   D,  C              : #
              MOV   E,  C              : #(DE)=0
              XCHG                     : #
              SHLD  TEST               : #
              SHLD  TEST + 2           : #
              SHLD  COMP               : #
              SHLD  COMP + 2           : CLEARS VARIABLES
              STA   MAJ                : #
              CALL  CMPAR              : COMPARES (DE)&(HL)... 0 & DY


              JZ    SEC5               : DY = ZERO
              XCHG                     : #
              LHLD  DX                 : #
              XCHG                     : #
              CALL  CMPAR              : # ... 0&DX
              JZ    SEC5               : DX=ZERO
      :
              LHLD  DY                 : #
              CALL  CMPAR              : #
              JC    HISEC              : DX<DY,  M>1
      :
              MVI   A,  1              : DX>=DY,  M<=1
              STA   MAJ                : MAJOR = 1
      :
              LDA   DIAG               : #
              ANI   0101B              : COMPUTES HORV
              STA   HORV               : #
              DAD   H                  : DY*2
              CALL  CMPAR              : #
              JC    SEC2               : DX<2*DY,  .5<M<1
              JMP   SEC1               : DX>=2*DY,  0<M<.5
      :
HISEC EQU   $                         : #
              LDA   DIAG               : #
              ANI   1010B              : COMPUTES HORV
              STA   HORV               : #
              XCHG                     : #(DE)=DY,  (HL)=DX
              DAD   H                  : DX*2
              CALL  CMPAR              : #
              JC    SEC3               : DY<2*DX,  1<M<2
```

```
          JMP   SEC4          ; DY>=2*DX,  M>2
    :
  SEC5  EQU   $             ; #
        XRA   A
        STC                 ; A IS 0
                              :+*
  LOOP5 EQU   $             ; #
        RAL
        DCR   B             ; LOOP COMPUTES
        JNZ   LOOP5         ; # PLOT CHAR
        STA   HORV          ; #   FROM QUADRANT (B)
        MVI   C, 4          ; #
        JMP   SDRIV         ; #
    :
  SEC4: INR   C             ; #
  SEC3: INR   C             ; #
  SEC2: INR   C             ; #
  SEC1: LDA   DIAG          ; PROCESS QUADS 3,  4 SEPARATELY
        ANI   1000B         ; #
        JNZ   QAD34         ; #
        MOV   A, C          ; #
        JMP   SDRIV         ; #
    :
  QAD34:MVI   A,  3         ; #
        SUB   C             ; #
    :
  SDRIV EQU   $             ; #
        STA   DRIVR         ; #
    :
    :                              LINSET
        LHLD  DY            ; #
        XCHG                ; #
        LHLD  DX            ; #(DE)=DY,  (HL)=DX
        LDA   MAJ           ; #
        DCR   A             ; #


        JNZ   XMAJ          ; PFOCESS ABS(SLOPE)>1 SEPARATELY
        XCHG                ; #
  XMAJ  EQU   $             ; #
        SHLD  DXI           ; #
        SHLD  IDIFF         ; #
        MOV   A, C          ; #
        MOV   B, H          ; #
        MOV   C, L          ; #
        DAD   H             ; #
        XCHG                ; #(DE)=2*DXI,  (HL)=DXJ
        SHLD  DXJ           ; #
        SHLD  JDIFF         ; #
    :
        SUI   4             ; A=OLD C (SECTOR)
        JZ    XSDRV         ; TRAP OUT AXES
        PUSH  H             ; #(BC)=IDIFF,  ((SP))=JDIFF
        LDA   DRIVR         ; #
        ANI   1             ; PROCESS DRIVERS 1,  3 SEPARATTELY
        JNZ   LNTST         ; #
        SHLD  COMP          ; COMP=JDIFF
        XCHG                ; #
        SHLD  TEST          ; #(DE)=DXJ,  (HL)=2*DXI
                            ; #          TEST=2*IDIFF
    :
  LNTST EQU   $             ; #
        XRA   A             ; CLEAR A
        LXI   H,  DXI       ; #
        ORA   M             ; #
        INX   H             ; #
        ORA   M             ; #
        INX   H             ; #
        ORA   M             ; #
        INX   H             ; #
        ORA   M             ; #
        JNZ   LNCMP         ; IF((DXI.OR.DXJ).EQ.0)
        POP   D             ; CLEARS STACK
                              :++
        JMP   WRAP          ; RETURNS TO MONITOR
                              :++
                              :++
  LNCMP EQU   $             ; #                                72
```

```
;                                (TEST-COMP)
        LHLD  TEST + 2        ; #
        XCHG                  ; #
        LHLD  COMP + 2        ; #
        CALL  CMPAR           ; #
        JC    PHORV           ; #
        JNZ   PDIAG           ; #
                             ; #**
        LHLD  TEST           ; #
        XCHG                  ; #
        LHLD  COMP           ; #
        CALL  CMPAR           ; #
        JC    PHORV           ; #
        JZ    EQTST           ; #
;
PDIAG   EQU   $              ; #
        LDA   DIAG           ; #
        CALL  MOVE           ; SEND TO PLOTTER
        LHLD  OXI            ; #
        DCX   H              ; DXI=DXI-1
        SHLD  DXI            ; #
        LHLD  COMP           ; #
        POP   D

        PUSH  D              ; BUT LEAVE ON STACK
        DAD   D              ; #
        SHLD  COMP           ; #
        JNC   PWRAP          ; #
;
        LHLD  COMP + 2       ; #
        INX   H              ; #
        SHLD  COMP + 2       ; #
        JMP   PWRAP          ; #
EQTST   EQU   $              ; #
        LDA   DRIVR          ; #
        DCR   A              ; #
        JZ    PDIAG          ; #
        POP   D              ; #
        PUSH  D              ; BUT LEAVE ON STACK
        MOV   H,  B          ; #
                             ; #**
        MOV   L,  C          ; #
        DAD   B              ; #
        CALL  CMPAR          ; #(DE)=JDIFF,   (HL)=2*IDIF
        LDA   DRIVR          ; #
        JNZ   TSTD1          ; #
        CPI   3              ; #
        JZ    PDIAG          ; #
        JMP   PHORV          ; #
TSTD1   EQU   $              ; #
        DCR   A              ; #
        JM    PDIAG          ; #
;
PHORV   EQU   $              ; #
        LDA   HORV           ; #
        CALL  MOVE           ; SEND TO PLOTTER
PWRAP   EQU   $              ; #
        LHLD  DXJ            ; #
        DCX   H              ; #
        SHLD  DXJ            ; #
        LHLD  TEST           ; #
        DAD   B              ; CALC. TEST
        SHLD  TEST           ; #
        JNC   LNTST          ; #
        LHLD  TEST + 2       ; #
        INX   H              ; #
        SHLD  TEST + 2       ; #
        JMP   LNTST          ; #
;
XSDRV   EQU   $              ; #
        DAD   B              ; ADD (BC)=IDIFF TO (HL)=JDIFF
        LDA   HORV           ; #
        MOV   C,  A          ; LOAD C WITH HORV
PAXIS   EQU   $              ; #
        MOV   A,  C          ; #
        CALL  MOVE           ; SEND TO PLOTTER
        DCX   H              ; #
        ORA   L              ; A WAS CLEAR
```

73

```
            JNZ    PAXIS
            ORA    H
            JNZ    PAXIS
                                        ;**
                                        ;**
                                        ;**
    WRAP    EQU    $
            POP    H              ; RESTORE
            POP    D
            POP    B
                                        ;**


            RET
    :
    :
    YCOMP   EQU    $              ; #
            CALL   DPSUB          ; COMPUTE DX
            SHLD   DX
                                        ;**
            LHLD   CURRY          ; #
            XCHG                  ; #
            LHLD   ENDY
            SHLD   CURRY          ; #
            XCHG
            CALL   CMPAF          ; #
            MVI    A,  DICMP      ; LOADS A WITH 10010001B
            RET                   ; #
                                        ;**
    CHR33   EQU    $              ; CHAR: A
            DB     1050
            DB     1160
            DB     1360
            DB     1450
            DB     1400
            DB     0030
            DB     3430
                                        ;**
    CHR34   EQU    $              ; CHAR: B
            DB     1060
            DB     1360
            DB     1450
            DB     1440
            DB     1330
            DB     1030
            DB     0330
            DB     1420
            DB     1410
            DB     1300
            DB     3000
                                        ;**
    CHR35   EQU    $              ; CHAR: C
            DB     0450
            DB     1360
            DB     1160
            DB     1050
            DB     1010
            DB     1100
            DB     1300
            DB     3410
                                        ;**
    CHR36   EQU    $              ; CHAR: D
            DB     1060
            DB     1360
            DB     1450
            DB     1410
            DB     1300
            DB     3000
                                        ;**
    CHR37   EQU    $              ; CHAR: E
            DB     1060
            DB     1460
            DB     0230
            DB     1030
```

74

```
              DB    0000
              DB    3400
                                      ;**
      :
      CHR38 EQU   $                   : CHAR: F
              DB    1060
              DB    1460
              DB    0230
              DB    3030              ;**
      :
      CHR39 EQU   $                   : CHAR: G
              DB    0230
              DB    1430
              DB    1410
              DB    1300
              D9    1100
              DB    1010
              DB    1050
              DB    1160
              DB    1360
              DB    3450
                                      ;**
      :
      CHR40 EQU   $                   : CHAR: H
              DB    1050
              DB    0030
              DB    1430
              DB    0460
              DB    3400
                                      ;**
      :
      CHR41 EQU   $                   : CHAR: I
              DB    1400
              DB    0460
              DB    1060
              DB    0260
              DB    3200
                                      ;**
      :
      CHR42 EQU   $                   : CHAR: J
              DB    0020
              DB    1010
              DB    1100
              DB    1300
              D9    1410
              DB    3460
                                      ;**
      :
      CHR43 EQU   $                   : CHAR: K
              DB    1060
              DB    0020
              DB    1460
              DB    0130
              DB    3400
                                      ;**
      :
      CHR44 EQU   $                   : CHAR: L
              DB    0060
              DB    1000
              DB    3400
                                      ;**
      :
      CHR45 EQU   $                   : CHAR: M
              DB    1060

              DB    1230
              DB    1460
              DB    3400
                                      ;**
      :
      CHR46 EQU   $                   : CHAR: N
              DB    1060
              DB    1400
              DB    3460
                                      ;**
```

75

```
        :
CHR47 EQU    $                   : CHAR: O
      DB     0010
      DB     1050
      DB     1160
      DB     1360
      DB     1450
      DB     1410
      DB     1300
      DB     1100
      DB     3010
                                  :**
        :
CHR48 EQU    $                   : CHAR: P
      DB     1060
      DB     1360
      DB     1450
      DB     1440
      DB     1330
      DB     3030
                                  :**
        :
CHR49 EQU    $                   : CHAR: Q
      DB     0010
      DB     1050
      DB     1160
      DB     1360
      DB     1450
      DB     1410
      DB     1300
      DB     1100
      DB     1010
      DB     0310
      DB     3400
                                  :**
        :
CHR50 EQU    $                   : CHAR: R
      DB     1060
      DB     1360
      DB     1450
      DB     1440
      DB     1330
      DB     1030
      DB     0130
      DB     3400
                                  :**
        :
CHR51 EQU    $                   : CHAR: S
      DB     0010
      DB     1100
      DB     1300
      DB     1410
      DB     1420
      DB     1330
      DB     1130


      DB     1040
      DB     1050
      DB     1160
      DB     1360
      DB     3450
                                  :**
        :
CHR52 EQU    $                   : CHAR: T
      DB     0060
      DB     1460
      DB     0260
      DB     3200
                                  :**
        :
                                  :**
CHR53 EQU    $                   : CHAR: U
      DB     0060
      DB     1010
      DB     1100
      DB     1300
      DB     1410
```

76

```
            DB    3460
                                    ;**
CHR54 EQU   $               ; CHAR: V
      DB    0060
      DB    1200
      DB    3460
                                    ;**
CHR55 EQU   $               ; CHAR: W
      DB    0060
      DB    1060
      DB    1230
      DB    1400
      DB    3460
                                    ;**
CHR56 EQU   $               ; CHAR: X
      DB    1460
      DB    0060
      DB    3400
                                    ;**
CHR57 EQU   $               ; CHAR: Y
      DB    0060
      DB    1230
      DB    1460
      DB    0230
      DB    3200
                                    ;**
CHR58 EQU   $               ; CHAR: Z
      DB    0060
      DB    1460
      DB    1000
      DB    3400
                                    ;**
CHR59 EQU   $               ; CHAR: [
      DB    0460
      DB    1260
      DB    1200
      DB    3400


                                    ;**
CHR60 EQU   $               ; CHAR: \
      DB    0060
      DB    3400
                                    ;**
CHR61 EQU   $               ; CHAR: ]
      DB    1200
      DB    1260
      DB    3060
                                    ;**
CHR62 EQU   $               ; CHAR: ^
      DB    0030
      DB    1260
      DB    3430
                                    ;**
CHR63 EQU   $               ; CHAR: _
      DB    3400
                                    ;**
CHR64 EQU   $               ; CHAR:
      DB    0160
      DB    3240
                                    ;**
CHR65 EQU   $               ; CHAR:
      DB    0460
      DB    1350
      DB    1340
      DB    1230
      DB    1320
```

77

```
        DB    1310
        DB    3400
                            ;**
  :
CHR66 EQU   $              ; CHAR:
        DB    0250
        DB    3210
                            ;**
  :
CHR67 EQU   $              ; CHAR:
        DB    0100
        DB    1210
        DB    1220
        DB    1330
        DB    1240
        DB    1250
        DB    3160
                            ;**
  :
CHR68 EQU   $              ; CHAR:
        DB    0020
        DB    1130
        DB    1310
        DB    3420
                            ;**
  :
CHR69 EQU   $              ; CHAR:   BLANK
        DB    2000
                            ;**
                            ;**
END2   EQU   $-1


                            ;****
                            ;****
                            ;****
        ORG   1000H
ENDX:  DW    0             ; #
ENDY:  DW    0             ; #
CURRX:DW    0             ; #
CURRY:DW    0             ; #
ORIGX:DW    0             ; #
ORIGY:DW    0             ; #
DX:    DW    0             ; #
DY:    DW    0             ; #
DXI:   DW    0             ; #
DXJ:   DW    0             ; #
IDIFF:DW    0             ; #
JDIFF:DW    0             ; #
  :
DIAG:  DB    0             ; #
HORV:  DB    0             ; #
ORIVR:DB    0             ; #
MAJ:   DB    0             ; #
SCALE:DB    0             ;
DIRN:  DB    0             ;
PSTAT:DB    0             ; PEN STATUS WORD
DSTAT EQU   1             ; FLAG INDICATES PEN DOWN
USTAT EQU   0             ; FLAG INDICATES PEN UP
  :**
TEST:  DW    0             ; #
        DW    0             ; #
COMP:  DW    0             ; #
        DW    0             ; #
                            ;**
PNTER:DW    0
                            ;**
BUFLN:DW    0
                            ;**
BUFER EQU   $              ; ORIGIN OF BUFFER
LSTND EQU   $-1
        END
```

DEPARTMENT OF DEFENSE

Director
Defense Advanced Rsch. Proj. Agency
    Attn: LTC W.A. Whitaker

Director
Defense Nuclear Agency
    Attn: TITL Tech. Library (3 copies
    Attn: TISI Archives
    Attn: RAAE Harold C. Fitz, Jr.
    Attn: RAAE Lt/Col. McKechney
    Attn: RAAE Capt. Peter Lunn
    Attn: RAAE Dr. Patrick Crowley

Dir. of Defense Rsch. & Engineering
Department of Defense
    Attn: DD/S&SS (OS) Daniel Brockway

Commander
Field Command
Defense Nuclear Agency
    Attn: FCPR

Chief Livermore Division
FLD Command DNA
    Attn: FCPRL

DEPARTMENT OF THE ARMY

Commander/Director
Atmospheric Sciences Laboratory
U.S. Army Electronics Command
    Attn: DRSEL-BL-SY-A.F. Niles
    Attn: H. Ballard

Commander
Harry Diamond Laboratories
    Attn: DRXDO-NP, F.H. Wiminetz
        (2 Copies)

Superintendent
Naval Post Graduate School
    Attn: Rsch Rpts Librarian

Commander
Naval Intelligence Support Center
    Attn: Document Control

Director
BMD Advanced Technical Center
    Attn: ATC-T, M. Capps
    Attn: ATC-O, M. Davies

Dep. Chief of Staff for Rsch. Dev & Accout.
Department of the Army
    Attn: MCB Division
    Attn: DAMA-CSZ-C
    Attn: DAMA-WSZC

Director
U.S. Army Ballistic Rsch. Labs.
    Attn: John Mester
    Attn: Tech. Library

Commander
U.S. Army Electronics Command
    Attn: Inst. for Expl. Research
    Attn: Weapons Effects Section

Commander
CORADCOM
    Attn: PP-Library
    Attn: DRDCO-COM-D

DEPARTMENT OF THE NAVY

Commander
Naval Oceans Systems Center
    Attn: Code 2200 William Moler

Director
Naval Research Laboratory
    Attn: Code 7712 D.P. McNut
    Attn: Code 6701 J.D. Brown
    Attn: Code 2600 Tech. Library
    Attn: Code 7175J C.Y. Johnson
    Attn: Code 6700 T.P. Coffey
    Attn: Code 7709 Wahab Ali
    Attn: Code 6780 D.F. Strobel
    Attn: Code 6780 P. Juluenne
    Attn: Code 67800 J. Fedder
    Attn: Code 6780 S. Ossakow
    Attn: Code 6707 J. Davis

Commander
Naval Surface Weapons Center
    Attn: Code WA 501 Navy NUC
           Programs Office
    Attn: Technical Library

DEPARTMENT OF THE AIR FORCE

AF Geophysics Laboratory, AFSC
    Attn: LKB, K.S.W. Champion
    Attn: OPR, A.T. Stair, Jr.
    Attn: OPR, P.G. Doyle
    Attn: OPR, R. Murphy
    Attn: LKO, R. Huffman
    Attn: OPR, J. Kennealy

AF Weapons Laboratory, AFSC
    Attn: Maj. Gary Ganong, DES

Commander
ASD
    Attn: ASD-YH-EX-LTC R. Leverette

SAMSO/AW
    Attn: SZJ Lt. Col. Doan

SAMSO/YN
    Attn: Maj. P. Sivgals

AFTAC
    Attn: Tech Library
    Attn: TD

HQ
Air Force Systems Command
    Attn: DLS
    Attn: Tech Library
    Attn: DLCAE
    Attn: DLTW
    Attn: DLXP
    Attn: SDR
    Attn: RDQ

US ENERGY RSCH. AND DEV. ADMIN

Division of Military Application
U.S. Energy Rsch. & Dev. Admin.
    Attn: DOC CON

Los Alamos Scientific Laboratory
    Attn: DOC CON for H.V. Argo
    Attn: DOC CON for M.B. Pongratz
    Attn: DOC CON for R. Brownlee
    Attn: Group AP-4, MS 567
    Attn: DOC CON for J. Zinn

University of California
Los Alamos Scientific Laboratory
    Attn: Librarian MS 362

Sandia Laboratories
    Attn: DOC CON for W.B. Brown,
            Org. 1353
    Attn: Tech Library Org. 3141

Argonne National Laboratory
Records Control
    Attn: DOC CON for D.W. Green
    Attn: DOC CON for LIR SVCS Rpts
            Sec.
    Attn: DON CON for G.T. Reedy

University of California
Lawrence Livermore Laboratory
    Attn: W.H. Duewer, L-262
    Attn: J. Chang, L-71

U.S. Energy Rsch. & Dev. Administration
Division of Headquarters Services,
Library Branch
    Attn: DOC CON for Class. Tech.
            Library

OTHER GOVERNMENT

Department of Transportation
Office of the Secretary
    Attn: S.C. Coroniti

NASA
Langley Station
    Attn: Tech. Library

NASA
Ames Research Center
    Attn: N-245-3 R. Whitten

Department of the Army
Bal. Missl. Def. Adv. Tech. Ctr.
    Attn: W.O. Davies

Federal Aviation Administration
    Attn: HAPP/AEQ-10/James W. Rogers

Central Intelligence Agency
    Attn: ED/SI RM 5G48 HQ Bldg.
    Attn: NED/OS I-2G4R HQS

Department of Commerce
National Bureau of Standards
    Attn: Sec. Officer for M. Krauss
    Attn: Sec. Off. for L.H. Gevantman

National Oceanic & Atmospheric Admin.
Environmental Research Laboratories
Department of Commerce
    Attn: G. Reid
    Attn: E. Ferguson
    Attn: F. Fehsenfeld

DEPARTMENT OF DEFENSE CONTRACTORS

Science Applications, Inc.
    Attn: D.G. Hopper

Aero-Chem Research Laboratories, Inc.
    Attn: A. Fontign
    Attn: H. Pergament

Aerodyne Research, Inc.
    Attn: F. Bien
    Attn: M. Camac

Aerospace Corporation
    Attn: N. Cohen
    Attn: H. Mayer
    Attn: R.J. McNeal
    Attn: T.D. Taylor
    Attn: J. Reinheimer
    Attn: R.D. Rawcliffe
    Attn: R. Herm
    Attn: P. Kisliuk

Battelle Memorial Institute
    Attn: H.L. LaMuth
    Attn: STOIAC

Brown Engineering Company, Inc.
    Attn: N. Passino

General Research Corporation
    Attn: D. Jones
    Attn: J. Ise, Jr.

California At Riverside, University of
    Attn: J.N. Pitts, Jr.
    Attn: A.M. Winer

California At San Diego, University of
    Attn: S.C. Lin

California University of Berkley
    Attn: Sec. Off. for H. Johnston
    Attn: Sec. Officer for Dept of
        Chem., H.L. Strauss

Calspan Corporation
    Attn: C.E. Treanor
    Attn: J.M. Grace
    Attn: M.G. Dunn
    Attn: W. Wruster

University of Colorado
Astro-Geophysics
    Attn: J.B. Pearce

Colorado, University of
Office of Contracts and Grants
    Attn: G.M. Lawrence, LASP

Concord Sciences
    Attn: E.A. Sutton

University of Denver
Space Science Laboratory
    Attn: B. Van Zyl

University of Denver
Denver Research Laboratory
    Attn: Sec Officer for D. Murcray

AVCO-Everett Research Laboratory Inc.
    Attn: Tech. Library
    Attn: C.W. Von Rosenberg, Jr.

General Electric Company
Space Division
    Attn: M.H. Bortner, Space Sci. Lab
    Attn: J. Burns
    Attn: F. Alyea
    Attn: P. Zavitsands
    Attn: R.H. Edsall
    Attn: T. Baurer

Geophysical Institute
University of Alaska
    Attn: J.S. Wagner
    Attn: Neal Brown

Lowell University of
Center for Atmospheric Research
    Attn: G.T. Best

Lockheed Missiles and Space Company
    Attn: J. Kumer, Dept 52-54
    Attn: J.B. Cladis, Dept 52-12, B202
    Attn: B.M. McCormac, Dept 52-54
    Attn: T. James, Dept 52-54
    Attn: M. Walt, Dept 52-10
    Attn: R.D. Sears, Dept 52-54

Institute for Defense Analysis
    Attn: E. Bauer
    Attn: H. Wolfhard

Mission Research Corporation
    Attn: D. Archer
    Attn: D. Fischer
    Attn: M. Scheibe
    Attn: D. Sappenfield
    Attn: D. Sowle

Photometrics, Inc.
    Attn: I.L. Kofsky

Berkeley Research Associates
    Attn: J.B. Workman

Physical Dynamics, Inc.
    Attn: A. Thomas

General Electric Company
Tempo-Center for Advanced Studies
    Attn: DASAIC
    Attn: W.S. Knapp
    Attn: T. Stephens
    Attn: D. Chandler
    Attn: V.R. Strull

Physics International Company
    Attn: DOC CON for Tech Library

Pittsburg, University of the Comwlth
System of Higher Education
    Attn: W.L. Fite
    Attn: M.A. Biondi
    Attn: F. Kaufman

R&D Associates
    Attn: R. Latter
    Attn: R.G. Lindgren
    Attn: B. Gabbard
    Attn: R. Lelevier
    Attn: A.L. Latter
    Attn: F. Gilmore
    Attn: H.J. Mitchell

Rand Corporation
    Attn: C. Crain

Science Applications, Inc.
    Attn: D.A. Hamlin
    Attn: D. Sachs
    Attn: D.G. Hopper

Stanford Research Institute International
    Attn: M. Baron
    Attn: W.G. Chesnut

Technology International Corporation
    Attn: W.P. Boquist

United Technologies Corporation
    Attn: R.H. Bullis

Utah State University
    Attn: D. Baker
    Attn: K. Baker
    Attn: C. Wyatt
    Attn: A. Steed

Physical Science, Inc.
    Attn: K. Wray
    Attn: R.L. Taylor
    Attn: G. Caledonia

Commander
Rome Air Development Center
    Attn: OSCA, J.J. Simons

Steward Radiance Laboratory
    Attn: R. Huppi

Boston College
Space Data Analysis Laboratory
    Attn: E.R. Hegblom
    Attn: W.F. Grieder

Forrestial Campus Library
Princeton University
    Attn: Librarian

Visidyne, Inc.
    Attn: H. Smith
    Attn: J.W. Carpenter
    Attn: T.C. Degges
    Attn: C. Humphrey
    Attn: W.P. Reidy

Wayne State University
    Attn: R.H. Kummler
    Attn: W.E. Kaupplia

DATE
FILMED

—8